

AD-A113 044

CLEMSON UNIV SC DEPT OF MATHEMATICAL SCIENCES

F/8 9/2

BASKET METHOD FOR SELECTING BALANCED SAMPLES. PART III. COMPUTE--ETC(U)

NOV 81 K T WALLENIS, S L BENZ

N00014-75-C-0451

NL

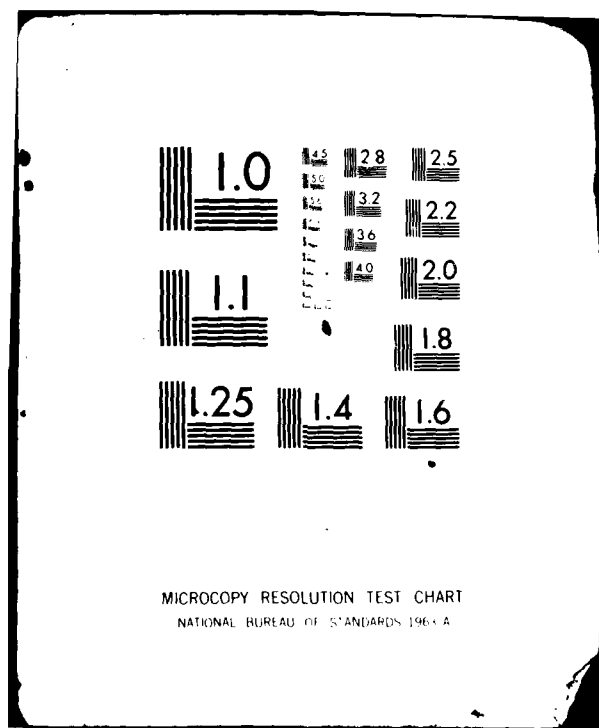
UNCLASSIFIED N134

1 10 1  
2 10 1  
3 10 1



END  
DATE  
FILMED

5-82  
DTIC



AD A11 3044

DEPARTMENT  
OF  
MATHEMATICAL  
SCIENCES

CLEMSON UNIVERSITY  
Clemson, South Carolina



DTIC FILE COPY

DTIC  
ELECTE  
APR 5  
H

DISTRIBUTION STATEMENT A  
Approved for public release  
Distribution Unlimited

82 04 0, 026

## **DISCLAIMER NOTICE**

**THIS DOCUMENT IS BEST QUALITY  
PRACTICABLE. THE COPY FURNISHED  
TO DTIC CONTAINED A SIGNIFICANT  
NUMBER OF PAGES WHICH DO NOT  
REPRODUCE LEGIBLY.**

(12)

BASKET METHOD FOR SELECTING BALANCED

SAMPLES - PART III:

COMPUTER SOURCE PROGRAMS

K. T. Wallenius  
and  
Stephen L. Benz\*

Department of Mathematical Sciences  
Clemson University

Technical Report #374

November 1981

N134

DTIC  
1982

This work was supported in part by the Office of Naval Research  
under Contract N00014-75-C-0451.

\*Computer Science Department, Clemson University, Clemson, SC 29631.

STATEMENT A  
Public release;  
unlimited

## INTRODUCTION

In this third of a series of three documents describing the basket method of sampling, source computer codes are provided for the convenience of users who may want to modify the programs to meet special local needs. The programming language used is the General Electric Mark III version of FORTRAN. These source programs and associated load modules are available to all users through the GE/USAF Copper Impact Time Sharing System.

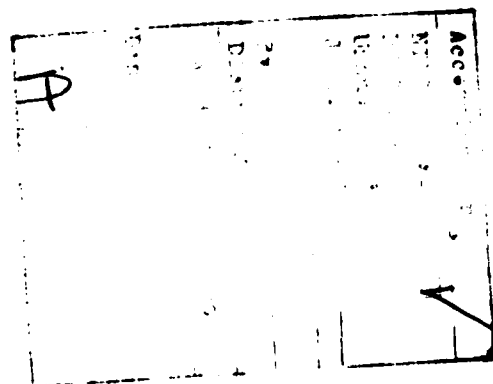
The programs can be run using a load module driver named BASKET\* with the GE system command:

### RUN BASKET\*

Source programs written in IBM FORTRAN IV are also available from the authors.

The source listing of four main programs and their associated subroutines are contained in this document. They are:

- BASKET: A driver program which calls the appropriate load modules as specified by the user.
- BASKSIM: Program used during the perspective analysis.
- BASSEL: Program for basket formation to be used in actual application.
- BASKAWARD: Program which prices the unsampled proposed based on the results of negotiating the proposals in the selected basket.



BASKET 17:25PDT 09/19/01

```
10 INTEGER AMS
11 CALL IFRK(1)
20 1 PRINT,"WHICH OF THE FOLLOWING WOULD YOU LIKE TO RUN:"
30 PRINT," "
40 PRINT," 1) BASKET SIMULATION MODEL"
50 PRINT," 2) BASKET SELECTION MODEL"
60 PRINT," 3) AWARD MODEL"
70 PRINT," 4) TERMINATE"
80 PRINT," "
90 PRINT," "
100 PRINT,"INPUT 1,2, OR 3"
110 PRINT," "
110 INPUT,AMS
120 IF(AMS.NE.1.AND.AMS.NE.2.AND.AMS.NE.3.AND.AMS.NE.4)GOTO 1
130 IF(AMS.EQ.1)CALL SYSTEM('/BASKSIM#')
140 IF(AMS.EQ.2)CALL SYSTEM('/BASSEL#')
150 IF(AMS.EQ.3)CALL SYSTEM('/BACAWRD#')
160 IF(AMS.EQ.4)STOP1
170 END
```

BACKOUT 15:00 EDT 01/10/81

```

1  OPTION PRTP
1000 MAIN PROGRAM
110    CALL DISPLAY
120    CALL INPUT
130    1  CALL PROCYM(02,03)
140    CALL LIST(01)
150    2  CALL SIMULA(01)
160    2  CALL PRICE(01)
170    STOP
180    END
1900
2000
210    SUBROUTINE DISPLAY
220    WRITE("      ",1)
230    1  FORMAT(' ',76('*'))/' ',74X,'*'/ ' ',25X,'PERSPECTIVE ANALYSIS '
240    1,'CF',26X,'*'/ ' ',30X,'BASKET METHOD',31X,'*'/ ' ',12X,'APPLIED '
250    2,'TO HISTORICAL BID/NEGOTIATED RELATIONSHIPS',12X,'*'/ ' ',74X
260    1,'*'/ ' ',25X,'SIMULATION AND PRICING',27X,'*'/ ' ',
270    2,25X,'VERSION IV - JULY 1981',27X,'*'/ ' ',74X
280    2,'*'/ ' ',14X,'THEORETICAL DEVELOPMENT - DR. K. T. WALLENIS',
290    3,15X,'*'/ ' ',10X,'ALGORITHM DESIGN AND IMPLEMENTATION -',
300    2,'DR. STEPHEN BENZ',10X,'*'/ ' ',74X,'*'/ ' ',29X,'CLEMSON',
310    2,'UNIVERSITY',25X,'*'/ ' ',31X,'CLEMSON S.C.',31X,'*'/ ' ',
320    2,74X,'*'/ ' ',76('*'))
330    RETURN
340    END
3500
3600
370    SUBROUTINE INPUT
380    CALL NUNCHR(01)
390    CALL CHRTYP
400    1  CALL BIDS
410    RETURN
420    END
4300
440    SUBROUTINE NUNCHR(*)
4500    READS NUMBER OF CHARACTER TYPES INTO - NCHAR.
4600
470    COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRSK(350,12)
480    INTEGER CHRNAM,CHRSK
490    1  WRITE("      ",2)
491    PRINT,""
500    2  FORMAT(' ',/, ' ENTER NUMBER OF CHARACTER TYPES.')
510    READ("      ",*) NCHAR
520    IF(NCHAR.EQ.0) NCHAR=1
530    IF(NCHAR.EQ.1) RETURN
540    IF(NCHAR.GT.0.AND.NCHAR.LE.12) RETURN
550    WRITE("      ",3) NCHAR
560    3  FORMAT(' ',/, ' *** ERROR ***  NUMBER OF CHARACTER TYPES,',13,',',
570    2/,15X,' IS NOT BETWEEN 1 AND 12 INCLUSIVE.')
580    WRITE("      ",4)
581    PRINT,""
590    4  FORMAT(' DO YOU WISH TO REENTER NUMBER OF CHARACTER TYPES?',
600    2' (YES OR NO)')
610    CALL ANSWER(01)
620    STOP
630    END
6400

```



```

0500
0600 SUBROUTINE CHRTYP
0700 READS CHARACTER TYPES INTO - CHRNAM(N,K), K=1,2.
0800
0900 COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
1000 INTEGER BLANK
1100 INTEGER CHRNAM,CHRESK
1200 DATA BLANK/' '/
1300 1 WRITE(" ",2) NCHAR
1400 PRINT,""
1500 2 FORMAT(' ',/, ' ENTER',I3,', ' 3 - CHARACTER ID'S FOR CHARACTER',
1600 3 ' TYPES, ONE PER LINE.')
1700 DO 3 J=1,NCHAR
1800 3 READ (" ",4) (CHRNAM(J,K),K=1,2)
1900 4 FORMAT(2A4)
2000 DO 5 M=1,NCHAR
2100 IF(CHRNAM(M,1).EQ.BLANK.AND.CHRNAM(M,2).EQ.BLANK) GO TO 6
2200 5 CONTINUE
2300 RETURN
2400 6 WRITE(" ",7) NCHAR
2500 7 FORMAT(' ',/, ' *** ERROR *** NUMBER OF CHARACTER TYPES DOES',/,
2600 8 &17X,' NOT EQUAL',I3,', ' , OR DATA NOT ENTERED IN PROPER FORMAT.')
2700 WRITE(" ",8)
2800 PRINT,""
2900 8 FORMAT(' DO YOU WISH TO REENTER CHARACTER TYPES? (YES OR NO)')
3000 CALL ANSWER(81)
3100 STOP
3200 END
3300
3400 SUBROUTINE BIDS
3500 READS DATA.
3600 BIDNUM(N) - BID PROPOSAL NUMBER.
3700 BIDPRI(N) - BID PRICE.
3800 BIDNEG(N) - NEGOTIATED PRICE.
3900 BIDTYP(N) - INTEGER VALUES FROM 1 TO 12 CORRESPONDING
4000 TO TYPE OF BID.
4100 CHNUMP(N) - NUMBER OF BIDS OF CHARACTER TYPE N.
4200
4300 COMMON /PCP/ NBIDS,CHNUMP(12)
4400 COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
4500 COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
4600 INTEGER CHNUMP,BIDNUM,BIDTYP,CHRNAM,CHRESK,PCP
4700 STRING FILENAME
4800 111 PRINT,"ENTER FILENAME"
4900 INPUT,FILENAME
5000 OPEN(FILENAME,STATUS="OLD",UNIT=9,ERR=100)
5100 GOTO 110
5200 100 PRINT,"FILE NOT FOUND"
5300 GOTO 111
5400 110 IF(NCHAR.GT.1) GO TO 2
5500 NERR=0

```

```

1080      DO 1 NBIDS=1,150
1090      BIDNUM(NBIDS)=NBIDS
1100      1  READ(9,*,END=11) BIDPRI(NBIDS),BIDNEG(NBIDS)
1110      GO TO 10
1120      2  DO 200 I=1,NCHAR
1130      200  CHNUMP(I)=0
1140      DO 3 NBIDS=1,350
1150      BIDNUM(NBIDS)=NBIDS
1160      3  READ(9,*,END=11) BIDPRI(NBIDS),BIDNEG(NBIDS),BIDTYP(NBIDS)
1170      IF(1.LE.BIDTYP(NBIDS).AND.BIDTYP(NBIDS).LE.NCHAR) GO TO 9
1180      PROP=BIDNUM(NBIDS)+NERR
1190      WRITE("      ",4) PROP,BIDTYP(NBIDS)
1200      4  FORMAT(' ',/, ' *** ERROR ENCOUNTERED ON PROPOSAL ',I4, '.'/
1210      2' THE TYPE OF PROPOSAL ',I2, ' DOES NOT MATCH WITH ANY OF',
1220      2' THOSE SUBMITTED.')
1230      NERR=NERR+1
1240      WRITE("      ",5)
1241  PRINT,""
1250      5  FORMAT(' DO YOU WISH TO OMIT THIS PROPOSAL FROM THE ANALYSIS',
1260      2' AND CONTINUE? (YES OR NO)')
1270      CALL ANSWER(83)
1280      3  CHNUMP(BIDTYP(NBIDS))=CHNUMP(BIDTYP(NBIDS))+1
1290      10  NBIDS=301
1300      11  NBIDS=NBIDS-1
1310      RETURN
1320      END
1330C
1340C
1350C
1360C
1370C
1380      SUBROUTINE PROGRAM(*,*)
1390      INTEGER SIML,PRIC,ANSWR,STP
1400      DATA LIST/'LIST'//,SIML/'SIML'//,PRIC/'PRIC'//,STP/'STOP'//
1410      1  WRITE("      ",2)
1411  PRINT,""
1420      2  FORMAT(//' ENTER "LIST" - TO LIST DATA',/,7X,' "SIML" - TO',
1430      2' RUN SIMULATION',/,7X,' "PRIC" - TO PERFORM PRICING',
1440      2/,7X,' "STOP" - TO END PROCESSING')
1450      READ("      ",3) ANSWR
1460      3  FORMAT(A4)
1470      IF(ANSWR.EQ.LIST) RETURN
1480      IF(ANSWR.EQ.SIML) RETURN 1
1490      IF(ANSWR.EQ.PRIC) RETURN 2
1500      IF(ANSWR.EQ.STP) STOP
1510      WRITE("      ",4) ANSWR
1520      4  FORMAT(' *** ERROR ***',A4,' IS NOT ONE OF ALLOWABLE COMMANDS.',
1530      2/,19X,' REENTER ONE OF FOLLOWING COMMANDS.')
1540      GO TO 1
1550      END
1560C

```

```

15700
15710 SUBROUTINE LIST(*)
15720 LIST=DATA.
15730 COMMON /PCP/ NBIDS,CHNUMP(12)
15740 COMMON /SID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
15750 COMMON /CHRCTP/ NCHAR,CHRNAM(12,2),CHRECK(350,12)
15760 INTEGER CHNUMP,BIDNUM,BIDTYP,CHRNAM,CHPRESK
15770 WRITE(" ",1) NBIDS
15780 1 FORMAT(//,' LISTING',//,' NUMBER OF BIDS IN DATA SET = ',I4,/)
15790 IF(NCHAR.EQ.1) GO TO 9
15800 WRITE(" ",2)
15810 2 FORMAT(' NUMBER OF BIDS PER CHARACTER TYPE.',/)
15820 DO 3 I=1,NCHAR
15830 3 WRITE(" ",4) I,(CHRNAM(I,J),J=1,2),CHNUMP(I)
15840 4 FORMAT(5X,I2,'.',2X,2A4,4X,I3)
15850 WRITE(" ",5)
15860 5 FORMAT(/,' DATA')
15870 WRITE(" ",6)
15880 6 FORMAT(10X,'BID PRICE',3X,'NEGOTIATED PRICE',3X,'CODE',/)
15890 DO 7 I=1,NBIDS
15900 7 WRITE(" ",9) I,BIDPRI(I),BIDNEG(I),BIDTYP(I)
15910 8 FORMAT(4X,I3,'.',1X,F10.3,4X,F10.3,3X,I2)
15920 RETURN
15930 9 WRITE(" ",5)
15940 WRITE(" ",10)
15950 10 FORMAT(10X,'BID PRICE',5X,'NEGOTIATED PRICE',/)
15960 DO 11 I=1,NBIDS
15970 11 WRITE(" ",12) I,BIDPRI(I),BIDNEG(I)
15980 12 FORMAT(5X,I3,'.',2X,F10.3,4X,F10.3)
15990 RETURN
16000 END
16010
16020 SUBROUTINE ANSWER(*)
16030 LOGICAL ARG
16040 INTEGER ANSWR,YES,NO
16050 DATA YES/'YES'/,NO/'NO'/
16060 1 READ(" ",2) ANSWR
16070 2 FORMAT(A2)
16080 IF(ANSWR.EQ.YES) RETURN
16090 IF(ANSWR.EQ.NO) STOP
16100 WRITE(" ",3)
16110 PRINT,""
16120 3 FORMAT(' REENTER "YES OR NO"')
16130 GO TO 1
16140 ENTRY ANSWR1(ARG)
16150 READ(" ",2) ANSWR
16160 ARG=.FALSE.
16170 IF(ANSWR.EQ.YES) ARG=.TRUE.
16180 RETURN
16190 END
16200
16210

```

```

02080 C
02090      SUBROUTINE SIMULA(*)
02095 C   PERFORMS SIMULATION RUN.
02100 C
02105      COMMON /RANGE/ NPASKL,NPASKH,NBASK
02110      COMMON /REPSEL/ NREP,RAND,IMBAL,NCODE
02115      LOGICAL RAND
02120      CALL CHORD(1)
02125      CALL SMINPU
02130      CALL SMINIT
02135      NPROMP=0
02140      DO 3 IREP=1,NREP
02145      NPROMP=NPROMP+1
02150      IF(NPROMP.LT.30) GO TO 100
02155      WRITE(3,200)
02160 200  FORMAT('ENTER "C" TO CONTINUE')
02165      READ(1,300) NPROPT
02170 300  FORMAT(A1)
02175      NPROMP=0
02180 100  IF(RAND) GO TO 1
02185      CALL CHOICE($2)
02190      1  CALL RANDOM
02195      2  DO 3 NBASK=NBASKL,NPASKH
02200      CALL BDORD
02205      CALL BASKT
02210      CALL SWAP
02215      3  CALL OVERAW
02220      CALL CHORD(3)
02225      CALL PCTMOM
02230      CALL RNSTAT
02235      DO 4 NBASK=NBASKL,NPASKH
02240      4  CALL HISTO
02245      CALL PETABL (21,NREP)
02250      RETURN
02255      END
02260 C

```

```

02265 C
02270 SUBROUTINE CHORD(K0)
02275 COMMON /POP/ NPIDS,CHNIMP(12)
02280 COMMON /BID/ BIDNUM(250),BIDPRT(250),BIDNEG(250),BIDTYP(250)
02285 COMMON /CHRCTP/ NCHAR,CHRNAM(12,2),CHRPBSK(250,12)
02290 INTEGER BIDTYP,BIDNUM,KK(12),ORIG(250)
02295 REAL SUM(12)
02300 IF(K0.EQ.1)GOTO 772
02305 DO 639 J=1,NPIDS
02310 639 BIDTYP(BIDNUM(J))=ORIG(BIDNUM(J))
02315 RETURN
02320 772 DO 546 I=1,NPIDS
02325 546 ORIG(BIDNUM(I))=BIDTYP(BIDNUM(I))
02330 DO 541 I=1,NCHAR
02335 541 SUM(I)=0.0
02340 DO 770 K=1,NBIDS
02345 770 SUM(BIDTYP(BIDNUM(K)))=SUM(BIDTYP(BIDNUM(K)))+BIDPRT(BIDNUM(K))
02350 K2=1
02355 DO 141 M=1,NCHAR
02360 141 BMAX=-10.**6
02365 DO 151 I=1,NCHAR
02370 151 IF(BMAX.GE.SUM(I))GOTO151
02375 KK(I)=K2
02380 BMAX=SUM(I)
02385 IM=I
02390 151 CONTINUE
02395 SUM(IM)=-10.**6
02400 141 K2=K2+1
02405 DO 557 M=1,NBIDS
02410 557 BIDTYP(BIDNUM(M))=KK(BIDTYP(BIDNUM(M)))
02415 RETURN
02420 END
02425 C
02430 C
02435 SUBROUTINE SMINPU
02440 CALL BSKRAM
02445 CALL NUMSUB
02450 CALL TYPSEL
02455 CALL NUMREP
02460 RETURN
02465 END
02470 C
02475 C
02480 SUBROUTINE BSKRAM
02485 C INPUTS LOW AND HIGH RANGE OF BASKETS INTO - NBASKL,NBASKH.
02490 COMMON /RANGE/ NBASKL,NBASKH,NBASK
02495 1 WRITE(" ",2)
02500 2 FORMAT(' ENTER RANGE FOR NUMBER OF BASKETS. ')
02505 READ(" ",*) NBASKL,NBASKH
02510 IF(2.LE.NBASKL.AND.NBASKL.LE.NBASKH.AND.NBASKH.LE.10) RETURN
02515 WRITE(" ",2)
02520 2 FORMAT(' *** ERROR *** RANGE OF BASKETS IS NOT BETWEEN 1,
02525 &'12 AND 10 INCLUSIVE',/,15X,'OR LOW RANGE IS LARGER THAN HIGH',
02530 &' RANGE.',/,15X,'DO YOU WISH TO REENTER? (YES OR NO)')
02540 CALL ANSWER(2)
02550 STOP
02560 END
02570 C

```

```

1400
1410 SUBROUTINE NUNSUB
1420 INPUTS NUMBER OF BIDS IN SUBPOPULATION INTO - NSUBP.
1430 COMMON /POP/ NBIDS,CHNUP(12)
1440 COMMON /SUBPOP/ NSUBP,CHNUMS(12)
1450 COMMON /RANGE/ NBASKL,NBASKH,NBASK
1460 INTEGER CHNUP,CHNUMS
1470
1480 1 WRITE(" ",2)
1490 PRINT,""
1500 2 FORMAT(' ENTER SUBPOPULATION SIZE.')
1510 READ(" ",*) NSUBP
1520 IF(NSUBP.LT.1000.AND.NSUBP.LE.NBIDS) RETURN
1530 WRITE(" ",2) NSUBP,NBASKH,NBASK
1540
1550 3 FORMAT(' *** ERROR *** VALUE OF SUBPOPULATION SIZE, ',I,
1560 ' IS NOT BETWEEN ',I3,/,15X,' AND ',I4,' INCLUSIVE.',/,15X.
1570 ' DO YOU WISH TO REENTER SUBPOPULATION SIZE? (YES OR NO)')
1580 CALL ANSWER(31)
1590 STOP
1600 END
1610
1620 SUBROUTINE TYPSEL
1630 INPUTS TYPE OF SUBPOPULATION SELECTION
1640 COMMON /REPSL/ NREP,RAND,INBAL,NCODE
1650 COMMON /CHRECT/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
1660 INTEGER CHRNAM,CHRESK
1670 LOGICAL RAND
1680 RAND=.TRUE.
1690 IF(NCHAR.EQ.1) RETURN
1700 7 WRITE(" ",1)
1710 PRINT,""
1720 1 FORMAT(' THERE ARE THREE SUBPOPULATION SELECTION OPTIONS:')
1730 2,/,10X,'1. SIMPLE RANDOM SAMPLING'
1740 2,/,10X,'2. STRATIFIED RANDOM SAMPLING - PROPORTIONAL ALLOCATION'
1750 2,/,10X,'3. STRATIFIED RANDOM SAMPLING - USER ALLOCATION'
1760 2,/, ' WHICH OPTION DO YOU PREFER? (1,2 OR 3)')
1770 READ(" ",*) NCODE
1780 GO TO (4,5,6), NCODE
1790 WRITE(" ",3) NCODE
1800 3 FORMAT(' ', '*** ERROR ***',I5,' IS NOT OPTION 1,2 OR 3.',/,
1810 ' DO YOU WISH TO REENTER? (YES OR NO)')
1820 CALL ANSWER(37)
1830 STOP
1840 4 RETURN
1850 5 RAND=.FALSE.
1860 CALL PROPSL
1870 RETURN
1880 6 RAND=.FALSE.
1890 CALL USERCH
1900 RETURN
1910 END
1920

```

```

00000
0000 SUBROUTINE NUMREP
01000 INPUTS NUMBER OF REPLICATION INTO - NREP.
0110 COMMON /PEPSEL/ NREP,RAND,INEAL,KCODE
0120 LOGICAL RAND
0130 1 WRITE(" ",2)
0131 PRINT,""
0140 2 FORMAT(' ENTER NUMBER OF REPLICATIONS. ')
0150 READ(" ",*) NREP
0160 IF(NREP.GE.1) RETURN
0170 WRITE(" ",3) NREP
0180 3 FORMAT(' *** ERROR *** NUMBER OF REPLICATIONS, ',I4,' IS NOT '
0190 ' GREATER THAN 0. ',/,15X,' DO YOU WISH TO REENTER NUMBER OF '
0200 ' REPLICATIONS? (YES OR NO) ')
0210 CALL ANSWER(31)
0220 STOP
0230 END
02400
02500
0260 SUBROUTINE CINIT
02700 INITIALIZES FREQ(I,J)=0 I=NBASKL,NBASKH, J=1,21
02800 BELOW(I)=0 I=NBASKL,NBASKH
02900 CUTOFR(I)=0 I=NBASKL,NBASKH
0300 COMMON /RANGE/ NBASKL,NBASKH,NBASK
0310 COMMON /RANDOM/ PERMUT(350),IX
0320 COMMON /HIST/ FREQ(21,10),BELOW(10),CUTOFR(10),CVRUNS(50,10)
0330 COMMON /ERROR/ PCTEPR,NEERROR(10),ERRSUM(10),ERRSQ(10),ERRMCN(10,2)
0340 INTEGER FREQ,BELOW,BSK,CUTOFR,PERMUT
0350 DO 1 BSK=NBASKL,NBASKH
0360 BELOW(BSK)=0
0370 CUTOFR(BSK)=0
0380 ERRSUM(BSK)=0.
0390 ERRSQ(BSK)=0.
0400 NEERROR(BSK)=0
0410 DO 1 J=1,21
0420 1 FREQ(J,BSK)=0
0430 ENTPY PRINIT
0440 IX=RRAND(0)
0450 RETURN
0460 END
04700
04800

```

```

15100
15101
15102
15103 SUBROUTINE PROCEL
15104 COMMON /POP/ NBIDS,CHNUNP(12)
15105 COMMON /SUBPOP/ NSUBP,CHNUMS(12)
15106 COMMON /CHCTER/ NCHAR,CHRNAM(12,2),CHRESK(250,12)
15107 INTEGER CHNUNP,CHNUMS,CHRNAM,CHRESK,ORDSZ(12)
15108 REAL XSMPST(12)
15109 NTOTSZ=0
15110 EPS=1.E-4
15111 XPROP=FLOAT(NSUBP)/FLOAT(NBIDS)
15112 DO 15 I=1,NCHAR
15113   XSMPST(I)=FLOAT(CHNUNP(I))*XPROP
15114   CHNUMS(I)=XSMPST(I)+ EPS
15115 15 NTOTSZ=NTOTSZ + CHNUMS(I)
15116   NDIFF=NSUBP-NTOTSZ
15117   IF(NDIFF.EQ.0) RETURN
15118   CALL ORDER(12,ORDSZ,XSMPST,NCHAR)
15119   KNT=1
15120   I=1
15121 7 IF(KNT.GT.NDIFF) RETURN
15122   IF(XSMPST(ORDSZ(I)).LE.0.0) GO TO 3
15123   CHNUMS(ORDSZ(I)) = CHNUMS(ORDSZ(I)) + 1
15124   KNT=KNT + 1
15125 3 I=I+1
15126   GO TO 7
15127 END
15128
15129 SUBROUTINE USERCH
15130 COMMON /POP/ NBIDS,CHNUNP(12)
15131 COMMON /SUBPOP/ NSUBP,CHNUMS(12)
15132 COMMON /CHCTER/ NCHAR,CHRNAM(12,2),CHRESK(250,12)
15133 INTEGER CHNUNP,CHNUMS,CHRNAM,CHRESK
15134 WRITE(" ",99)
15135 99 FORMAT(' ', 'STRATA COMPOSITION OF POPULATION',/)
15136 WRITE(" ",100) (CHRNAM(I,1),CHRNAM(I,2),CHNUNP(I),I=1,NCHAR)
15137 100 FORMAT(4(' ',A4,A4,1X,I3,2X))
15138 1 NPROP=0
15139 WRITE(" ",2) NCHAR,NSUBP
15140 2 FORMAT(/, ' ENTER ',I2, ' STRATA SAMPLE SIZES FOR SUBPOPULATION'
15141 &,/, ' SUM OF THE SIZES MUST EQUAL ',I3,/)
15142 DO 12 I=1,NCHAR
15143   WRITE(" ",3) (CHRNAM(I,J),J=1,2)
15144 3 FORMAT(' ',2A4)
15145 4 CONTINUE
15146 READ(" ",*) CHNUMS(I)
15147 IF(0.LE.CHNUMS(I).AND.CHNUMS(I).LE.CHNUNP(I)) GO TO 11
15148 WRITE(" ",5) (CHRNAM(I,J),J=1,2),CHNUNP(I)
15149 5 FORMAT(' *** ERROR *** STRATA SAMPLE SIZE FOR CHARACTER TYPE, '
15150 &,2A4, ' IS LESS THAN 0',/, 'EX', 'OR GREATER THAN NUMBER OF AIDS',
15151 &, ' OF THAT TYPE, ',I3, ' ',/, 'EX', 'DO YOU WISH TO REENTER',
15152 &, ' STRATA SAMPLE SIZE? (YES OR NO)')
15153 CALL INCHER(25)
15154 6 WRITE(" ",7)

```



```

4041 PRINT,""
4042      3 PRINT(' REENTER VALUE. ')
4043      10 TO 4
4044      11 NSUBP=NSUBP + CHNUMS(I)
4045      12 CONTINUE
4046      IF(NSUBP.EQ.NSUBP) RETURN
4047      WRITE(" ",13) NSUBP,NSUBP
4048      15 FORMAT(' *** ERROR *** SUM OF STRATA SAMPLE SIZES,',I4,
4049      1' DOES NOT EQUAL',/,15X,'SUBPOPULATION SIZE,',I4,'.',/,15X
4050      1'DO YOU WISH TO REENTER STRATA SAMPLE SIZES? (YES OR NO)')
4051      CALL ANSWER(11)
4052      STOP
4053      END
4054
4055
4056
4057
4058      SUBROUTINE CHOICE(*)
4059      SELECTS NSUBP BIDS ACCORDING TO THE PROPORTION NUMBERS INPUTED.
4060
4061      COMMON /POP/ NBIDS,CHNUMP(12)
4062      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
4063      COMMON /CHRCR/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
4064      INTEGER CHNUMP,CHNUMS,CHRNAM,CHRESK
4065      INTEGER CHARA
4066      CALL CHBASK(NBIDS)
4067      DO 1 CHARA=1,NCHAR
4068      CALL PERM(CHNUMS(CHARA),CHNUMP(CHARA))
4069      1 CALL SWITCH(CHRESK(1,CHARA),CHNUMP(CHARA))
4070      CALL BSKORD
4071      RETURN
4072      END
4073
4074
4075
4076
4077      SUBROUTINE RANDOM
4078      SELECTS THE SUBPOPULATION FROM THE PARENT POPULATION RANDOMLY.
4079
4080      COMMON /POP/ NBIDS,CHNUMP(12)
4081      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
4082      COMMON /RANDOM/ PERMUT(350),IX
4083      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
4084      INTEGER CHNUMP,CHNUMS,BIDNUM,BIDTYP,PERMUT
4085      CALL PERM(NSUBP,NBIDS)
4086      CALL SWITCH(BIDNUM(1),NBIDS)
4087      CALL CHBASK(NSUBP)
4088      CALL BSKORD
4089      RETURN
4090      END
4091
4092

```

```

45010
4510 SUBROUTINE PERM(STOP1,END)
4520 COMMON /RANDOM/ PERMUT(350),IX
4530 INTEGER PERMUT,COUNT,STOP1,END,END1
4540 COUNT=0
4550 DO 1 I=1,END
4560 1 PERMUT(I)=I
4570 END1=END-1
4580 DO 2 I=1,END1
4590 J=END-I+1
4600 K=IRAND(J)+I-1
4610 L=PERMUT(I)
4620 PERMUT(I)=PERMUT(K)
4630 PERMUT(K)=L
4640 COUNT=COUNT+1
4650 IF(COUNT.EQ.STOP1) RETURN
4660 2 CONTINUE
4670 RETURN
4680 END
4690C
4700C
4710 FUNCTION IRAND(J)
4720 COMMON /RANDOM/ PERMUT(350),IX
4730 INTEGER PERMUT
4740 RAND=RND(IX)
4750 IRAND=RAND*J+1
4760 RETURN
4770 END
4780C
4790C
4800C
4810 SUBROUTINE SWITCH(ARRAY,SIZE)
4820 INTEGER ARRAY(350),PERMUT,ARRAY1(350),SIZE
4830 COMMON /RANDOM/ PERMUT(350),IX
4840 DO 1 I=1,SIZE
4850 1 ARRAY1(I)=ARRAY(PERMUT(I))
4860 DO 2 I=1,SIZE
4870 2 ARRAY(I)=ARRAY1(I)
4880 RETURN
4890 END
4900C
4910C

```

```

49200
49300
49400
49500      SUBROUTINE CHBASK(BIDS)
49600      PUTS BIDS INTO BASKETS - CHRBSK(N,K) - BY TYPE.
49800
49700      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
49800      COMMON /RANGE/ NBASKL,NBASKH,NBASK
49900      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
50000      COMMON /BASK/ BASKET(10,350),BACK1(10,160),BTOT(10),BNEG(10)
50100      COMMON /CHRECT/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
50200      COMMON /REPSL/ NREP,RAND,IMBAL,NCODE
50300      INTEGER CHNUMS,BIDNUM,BIDTYP,BASKET,BACK1,CHRNAM,CHRESK
50400      INTEGER CHARA,CHRNUM(12),BIDS
50500      LOGICAL RAND
50600      IMBAL=0
50700      IF(NCHAR.EQ.1) GO TO 9
50800      DO 1 CHARA=1,NCHAR
50900      1  CHRNUM(CHARA)=0
51000      DO 2 NBID=1,BIDS
51100      NUMEID=BIDNUM(NBID)
51200      INDEX=BIDTYP(NUMEID)
51300      CHRNUM(INDEX)=CHRNUM(INDEX)+1
51400      2  CHRBSK(CHRNUM(INDEX),INDEX)=NUMEID
51500      IF(.NOT.RAND) RETURN
51600      DO 3 CHARA=1,NCHAR
51700      IF(CHRNUM(CHARA).GE.NBASKH) GO TO 8
51800      IMBAL=IMBAL+1
51900      3  CHNUMS(CHARA)=CHRNUM(CHARA)
52000      RETURN
52100      9  CHNUMS(1)=NSUBP
52200      DO 10 NBID=1,BIDS
52300      10  CHRESK(NBID,1)=BIDNUM(NBID)
52400      RETURN
52500      END
52600

```

```

50700
50800 SUBROUTINE BSKORD
50900 PUTS BIDS WITHIN EACH CHARACTER TYPE BASKET IN ORDER
51000 FROM LARGEST TO SMALLEST.
51100
51200 COMMON /SUBPOP/ NSUBP,CHNUMS(12)
51300 COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
51400 COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
51500 INTEGER CHNUMS,BIDNUM,BIDTYP,CHRNAM,CHRESK
51600 INTEGER CHARA,FIRST,RECENT,TEMP
51700 DO 5 CHARA=1,NCHAR
51800 RECENT=1
51900 LAST=CHNUMS(CHARA)
52000 IF(LAST.LE.1) GO TO 5
52100 1 FIRST=RECENT+1
52200 DO 2 J=FIRST, LAST
52300 JM1=J-1
52400 IF(BIDPRI(CHRESK(JM1,CHARA)).GE.BIDPRI(CHRESK(J,CHARA))) GO TO 2
52500 RECENT=JM1
52600 TEMP=CHRESK(JM1,CHARA)
52700 CHRESK(JM1,CHARA)=CHRESK(J,CHARA)
52800 CHRESK(J,CHARA)=TEMP
52900 2 CONTINUE
53000 IF(RECENT+1.EQ.FIRST) GO TO 5
53100 LAST=RECENT
53200 J=LAST
53300 3 JM1=J-1
53400 IF(BIDPRI(CHRESK(JM1,CHARA)).GE.BIDPRI(CHRESK(J,CHARA))) GO TO 4
53500 RECENT=J
53600 TEMP=CHRESK(JM1,CHARA)
53700 CHRESK(JM1,CHARA)=CHRESK(J,CHARA)
53800 CHRESK(J,CHARA)=TEMP
53900 4 J=J-1
54000 IF(J.GE.FIRST) GO TO 3
54100 IF(RECENT.LT.LAST) GO TO 1
54200 5 CONTINUE
54300 RETURN
54400 END
54500

```

```

5650C
5660C
5670C
5680C
5690C      SUBROUTINE BDORD
5700C      DETERMINES ORDER IN WHICH BIDS WILL BE PLACED IN GROUPS OF NBASK.
5710C      CREATES - BIDOR(N) - WHICH IS THE INDEX OF THE CHARACTERISTIC
5720C      OF THE NTH BID GROUPING.
5730C      NGRUP - NUMBER OF SUCH GROUPINGS.
5740C
5750C      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
5760C      COMMON /RANGE/ NBASKL,NBASKI,NBASK
5770C      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
5780C      COMMON /BIDORD/ BIDOR(100),NGROUP
5790C      COMMON /BASK/ BASKET(10,350),BASK1(10,100),BTOT(10),BNEG(10)
5800C      COMMON /CHRCR/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
5810C      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
5820C      INTEGER CHNUMS,BIDNUM,BIDTYP,BIDOR,BASKET,BASK1,CHRNAM,CHRESK
5830C      INTEGER CHARA
5840C      NCHAR1=NCHAR
5850C      NGRUP=0
5860C      NUMBID=0
5870C      IF(NCHAR.EQ.1) GO TO 4
5880C      DO 1 CHARA=1,NCHAR
5890C 1      NCHR(CHARA)=0
5900C      DO 2 N=1,400
5910C      IF(NUMBID.GE.NSUBP) RETURN
5920C      CALL MAXAVG
5930C      NGRUP=NGROUP+1
5940C      BIDOR(NGROUP)=MAX
5950C 2      CONTINUE
5960C      RETURN
5970C 4      NGRUP=NSUBP/NBASK
5980C      IF(NBASK*NGROUP.NE.NSUBP) NGRUP=NGROUP+1
5990C      DO 5 NGROUP=1,NGROUP
6000C 5      BIDOR(NGROUP)=1
6010C      NCHR(1)=NSUBP
6020C      RETURN
6030C      END
6040C

```

```

06050 C
06060 SUBROUTINE MAXAVG
06070 C CALCULATES AVERAGE OF THE NEXT NRASK BIDS OF EACH CHARACTERISTIC
06080 C TYPE, AND RETURNS THE INDEX OF THE CHARACTERISTIC TYPE HAVING THE
06090 C LARGEST AVERAGE.
06100 C NCHR(N) = INDEX OF POSITION TO BEGIN AVERAGE OF BIDS FOR
06110 C CHARACTERISTIC TYPE N.
06120 C
06130 COMMON /SUBPOP/ NSUBP,CHNUMS(12)
06140 COMMON /RANGE/ NRASKL,NRASKH,NRASK
06150 COMMON /BID/ BIDNUM(250),BIDPRI(250),BIDNEG(250),BIDTYP(250)
06160 COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHRRSK(250,12)
06170 COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
06180 INTEGER CHNUMS,BIDNUM,BIDTYP,CHNAM,CHRRSK
06190 INTEGER START,END
06200 MAX=1
06210 DO 2 N=1,NCHAR1
06220 AVG(N)=0
06230 START=NCHR(N)+1
06240 END=NCHR(N)+NRASK
06250 IF(END.GT.CHNUMS(N)) END=CHNUMS(N)
06260 IF(START.GT.END) GO TO 2
06270 DO 1 J=START,END
06280 1 AVG(N)=AVG(N)+BIDPRI(CHRRSK(J,N))
06290 AVG(N)=AVG(N)/(END-START+1)
06300 2 IF(AVG(N).LT.AVG(MAX)) GO TO 2
06310 MAX=N
06320 MAXBDS=END-START+1
06330 2 CONTINUE
06340 NUMBID=NUMBID+MAXPDS
06350 NCHR(MAX)=NCHR(MAX)+MAXPDS
06360 RETURN
06370 END
06380 C
06390 C

```

```

06400 C
06410 C
06420      SUBROUTINE BASKT
06440 C      PLACES BIDS INTO BASKETS ACCORDING TO BIDOR AND BASKET TOTALS.
06460 C
06480      COMMON /RANGE/ NBASKL,NBASKH,NBASK
06500      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
06520      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
06540      COMMON /BIDORD/ BIDOR(160),NGROUP
06560      COMMON /CHRCR/ NCHAR,CHRNAM(12,2),CHRBKS(350,12)
06580      COMMON /AVGORD/AVG(12),NCHR(12),NUMRID,NCHAR1,MAY
06600      INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR,CHRNAM,CHRBKS
06620      INTEGER START(12),UP1(10),ORJ,ORDER1(10),ORDER2(10),RIDORI
06640      DO 1 N=1,NBASK
06660      BTOT(N)=0
06680      1  UP1(N)=0
06690      DO 20 N=1,NCHAR
06700      20  START(N)=0
06710      12 6 I=1,NGROUP
06720      RIDORI=BIDOR(I)
06730      NUMLEF=NCHR(RIDORI)-START(RIDORI)
06740      CALL ORDER(10,ORDER1,BTOT,NBASK)
06750      IF(NUMLEF.LT.NBASK) GO TO 3
06760      DO 2 J=1,NBASK
06770      ORJ=ORDER1(J)
06780      INDEX=CHRBKS(START(RIDORI)+J,RIDORI)
06790      PRICE=BIDPRI(INDEX)
06800      BASKET(ORJ,I)=INDEX
06810      2  BTOT(ORJ)=BTOT(ORJ)+PRICE
06820      START(RIDORI)=START(RIDORI)+NBASK
06830      GO TO 6
06840      3  CALL UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
06850      DO 4 J=1,NUMLEF
06860      ORJ=ORDER2(J)
06870      INDEX=CHRBKS(START(RIDORI)+J,RIDORI)
06880      PRICE=BIDPRI(INDEX)
06890      BASKET(ORJ,I)=INDEX
06900      4  BTOT(ORJ)=BTOT(ORJ)+PRICE
06910      NUMLEF1=NUMLEF+1
06920      DO 5 J=NUMLEF1,NBASK
06930      ORJ=ORDER2(J)
06940 C      WHEN NUMBER OF BIDS LEFT OF ANY PARTICULAR CHARACTERISTIC TYPE
06950 C      IS LESS THAN NUMBER OF BASKETS, THE BIDS ARE PLACED INTO
06960 C      BASKETS BY THE USUAL METHOD AND THE UNFILLED BASKETS RECEIVE
06970 C      THE VALUE -1 INTO BASKET(N,K), THE INDICE ARRAY.
06980 C      FOR LATER CORRECTION OF BTOT(N).
06990      5  BASKET(ORJ,I)=-1
07000      6  CONTINUE
07010      RETURN
07020      END
07030 C

```

```

70400
70500 SUBROUTINE ORDER(N,ORDER1,STOT,NBASK)
70600 ORDER1=THE BASKET INDICES INTO = ORDER1 = FROM SMALLEST BASKET
70700 TO LARGEST BASKET.
70800
70900 INTEGER ORDER1(N)
71000 REAL STOT(N)
71100 DO 1 I=1,NBASK
71200 ORDER1(I)=I
71300 1 NBASK1=NBASK-1
71400 DO 2 I=1,NBASK1
71500 MIN=I
71600 NBASK2=N+1
71700 DO 2 K=NBASK2,NBASK
71800 IF(STOT(ORDER1(MIN)).LE.STOT(ORDER1(K))) GO TO 2
71900 ITEMP=ORDER1(MIN)
72000 ORDER1(MIN)=ORDER1(K)
72100 ORDER1(K)=ITEMP
72200 2 CONTINUE
72300 3 CONTINUE
72400 RETURN
72500 END
72600
72700
72800 SUBROUTINE UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
72900 DETERMINES WHICH BASKETS HAVE ONE MORE BID THAN OTHERS AND RETURNS
73000 - ORDER2 - THE NEW ORDER OF BASKET INDICES REFLECTING SMALLEST TO
73100 LARGEST AND NUMBER OF BIDS IN BASKETS.
73200
73300 INTEGER PORDER1(10),ORDER2(10),UP1(10),ENDJ,STARTJ
73400 STARTJ=0
73500 DO 1 I=1,NBASK
73600 IF(UP1(ORDER1(I)).EQ.1) GO TO 1
73700 STARTJ=STARTJ+1
73800 ORDER2(STARTJ)=ORDER1(I)
73900 1 CONTINUE
74000 IF(STARTJ.NE.0) GO TO 2
74100 DO 2 J=1,NBASK
74200 2 ORDER2(J)=ORDER1(J)
74300 GO TO 3
74400 3 IF(STARTJ.EQ.NBASK) GO TO 4
74500 ENDJ=STARTJ
74600 DO 4 I=1,NBASK
74700 IF(UP1(ORDER1(I)).EQ.0) GO TO 4
74800 ENDJ=ENDJ+1
74900 ORDER2(ENDJ)=ORDER1(I)
75000 4 CONTINUE
75100 5 IF(STARTJ.GE.NUMLEF) GO TO 10
75200 DO 6 J=1,NUMLEF
75300 IF(UP1(ORDER2(J)).EQ.0) UP1(ORDER2(J))=0
75400 6 IF(UP1(ORDER2(J)).EQ.1) UP1(ORDER2(J))=1
75500 NUMLEF1=NUMLEF+1
75600 DO 7 J=NUMLEF1,NBASK
75700 7 UP1(ORDER2(J))=0
75800 RETURN
75900 8 NUMLEF1=NUMLEF+1
76000 DO 9 J=NUMLEF1,NBASK
76100 9 UP1(ORDER2(J))=0
76200 10 DO 11 J=1,NUMLEF
76300 11 UP1(ORDER2(J))=1
76400 RETURN
76500 END
76600

```



```

7000 SUBROUTINE SWAP
7010 PERFORM A BID SWAP BETWEEN BASKETS TO BALANCE BASKET TOTALS.
7020 LGRSK - INDEX OF LARGEST BASKET WITH RESPECT TO BASKET TOTALS.
7030 SMBSK - INDEX OF SMALL BASKET.
7040
7050 COMMON /BID/ BIDNUM(350), BIDPRI(350), BIDNEG(350), BIDTYP(350)
7060 COMMON /BASK/ BASKET(10,350), BASK1(10,150), BTOT(10), BNEG(10)
7070 COMMON /RANGE/ NBASKL, NBASKH, NBASK
7080 COMMON /SWAP1/ LGRSK, LGIND, SMBSK, SMIND, STARTC(12), CHAR
7090 INTEGER BIDNUM, BIDTYP, BASKET, BASK1
7100 INTEGER LGRSK, LGIND, SMBSK, SMIND, STARTC, CHAR
7110 INTEGER ORDER1(10), BSK, SWITCH1, SWITCH2
7120 CALL SETUP
7130 1 CALL ORDER(10, ORDER1, BTOT, NBASK)
7140 DO 3 N=2, NBASK
7150 LGRSK=ORDER1(NBASK-N+2)
7160 NBASK1=NBASK-1
7170 DO 2 K=1, NBASK1
7180 SMBSK=ORDER1(K)
7190 DIFF=BTOT(LGRSK)-BTOT(SMBSK)
7200 IF(DIFF.EQ.0.) GO TO 3
7210 CALL DIFFNT(SWITCH2, DIFF)
7220 IF(SMIND.NE.0) GO TO 4
7230 2 CONTINUE
7240 3 CONTINUE
7250 RETURN
7260 4 DIFF=BIDPRI(BASK1(LGRSK, LGIND))-BIDPRI(BASK1(SMBSK, SMIND))
7270 BTOT(LGRSK)=BTOT(LGRSK)-DIFF
7280 BTOT(SMBSK)=BTOT(SMBSK)+DIFF
7290 CALL PLACE
7300 GO TO 1
7310 END
7320
7330 SUBROUTINE SETUP
7340 INITIALIZES - BASK1(N,K) - TO BE THE SAME AS - BASKET(N,K) - BUT
7350 WITH BIDS PLACED TOGETHER WITHIN BASKETS BY CHARACTERISTIC TYPE.
7360 NOTE: BIDS WILL ALSO BE IN ORDER FROM LARGEST TO SMALLEST
7370 WITHIN CHARACTER GROUPINGS SINCE BASKET WAS ORDERED.
7380
7390 COMMON /RANGE/ NBASKL, NBASKH, NBASK
7400 COMMON /BIDCRD/ BIDCR(150), NGROUP
7410 COMMON /BASK/ BASKET(10,350), BASK1(10,150), BTOT(10), BNEG(10)
7420 COMMON /CHRCR/ NCHAR, CHRNAM(12,2), CHRBSK(350,12)
7430 COMMON /SWAP1/ LGRSK, LGIND, SMBSK, SMIND, STARTC(12), CHAR
7440 INTEGER BIDCR, BASKET, BASK1, CHRNAM, CHRBSK
7450 INTEGER LGRSK, LGIND, SMBSK, SMIND, STARTC, CHAR
7460 INTEGER COUNT(12), POS, BSK
7470 DO 1 N=1, NCHAR
7480 1 COUNT(N)=0
7490 DO 2 N=1, NGROUP
7500 NUMB=BIDCR(N)
7510 2 COUNT(NUMB)=COUNT(NUMB)+1
7520 STARTC(1)=1

```

```

08230      DO 3 N=2,NCHAR
08230      STARTC(N)=COUNT(N-1)+STARTC(N-1)
08240      3 COUNT(N-1)=0
08250      COUNT(NCHAR)=0
08260      DO 5 N=1,NGROUP
08270      NUMP=BIDOR(N)
08280      POS=STARTC(NUMP)+COUNT(NUMP)
08290      DO 4 BSK=1,NBASK
08300      4 BASK1(BSK,POS)=BASKET(BSK,N)
08310      5 COUNT(NUMP)=COUNT(NUMP)+1
08320      RETURN
08330      END
08340      C
08350      C
08360      SUBROUTINE DIFFNT(DIFF,DIFF1)
08365      C DETERMINES IF/WHICH BID SWAPS FOR GIVEN BASKETS WILL PRODUCE A
08370      C BETTER BALANCE.
08375      C      LGIND - INDEX OF BID TO BE SWAPPED OUT OF LARGE BASKET.
08380      C      SMIND - INDEX OF BID TO BE SWAPPED OUT OF SMALL BASKET.
08385      C
08390      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
08395      COMMON /BIDORD/ BIDOR(160),NGROUP
08400      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
08405      COMMON /CHRCR/ NCHAR,CHRNAM(12,2),CHPSK(350,12)
08410      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
08415      INTEGER BIDNUM,BIDTYP,BIDOR,BASKET,BASK1,CHRNAM,CHPSK
08420      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
08425      INTEGER SWCH1,SWCH2,STRT,STP,CHARA,EQUAL
08430      EQUAL=0
08435      SMIND=0
08440      DIFF1=0
08445      DIFF12=DIFF/2.
08450      DIFF1D=ABS(DIFF1-DIFF12)
08455      DO 3 CHARA=1,NCHAR
08460      STPT=STARTC(CHARA)
08465      STP=NGROUP
08470      IF(CHARA.LT.NCHAR)STP=STARTC(CHARA+1)-1
08475      DO 2 SWCH1=STRT,STP
08480      IND1=BASK1(LGBSK,SWCH1)
08485      IF(IND1.EQ.-1) GO TO 2
08490      PRICE1=BIDPRI(IND1)
08495      DO 1 SWCH2=STRT,STP
08500      IND2=BASK1(SMBSK,SWCH2)
08505      IF(IND2.NE.-1.) GO TO 10
08510      DIFF2=PRICE1
08515      GO TO 12

```

```

08530      10  PRICE2=BIDPRI(IND2)
08535      DIFF2=PRICE1-PRICE2
08540      12  IF(DIFF2.LE.C.) GO TO 1
08545      IF(DIFF2.GE.DIFF) GO TO 2
08550      DIFF2D=ABS(DIFF2-DIFF12)
08560      IF(DIFF2D.LT.C) GO TO 14
08570      IF(DIFF2D.GT.DIFF1D) GO TO 2
08580      GO TO 15
08590      14  IF(DIFF2D.GT.DIFF1D) GO TO 1
08600      16  IF(DIFF2D.EQ.DIFF1D) GO TO 20
08610      EQUAL=0
08620      DIFF1=DIFF2
08630      LGIND=SWTCH1
08640      SMIND=SWTCH2
08650      CHAR=CHARA
08660      C    IF(DIFF2.EQ.DIFF/2.) RETURN
08670      GO TO 1
08680      20  IF(SMIND.EQ.C) GO TO 1
08690      EQUAL=EQUAL+1
08700      1    CONTINUE
08710      2    CONTINUE
08720      3    CONTINUE
08730      IF(EQUAL.EQ.C) RETURN
08740      WRITE(" ",20) EQUAL
08750      30  FORMAT(' ',I6,' OTHER EQUAL SWAPS.')
08760      RETURN
08770      END
08780      C
08790      C
08800      SUBROUTINE PLAGE
08810      C    SWAPS BIDS.
08820      C
08830      COMMON /BASK/ BASKET(10,350),BASK1(10,150),BTOT(10),BNEG(10)
08840      COMMON /SWAP1/LGBSK,LGIND,SMPSK,SMIND,STARTC(12),CHAR
08850      INTEGER BASKET,BASK1
08860      INTEGER LGBSK,LGIND,SMPSK,SMIND,STARTC,CHAR
08870      TEMP=BASK1(LGBSK,LGIND)
08880      BASK1(LGBSK,LGIND)=BASK1(SMPSK,SMIND)
08890      BASK1(SMPSK,SMIND)=TEMP
08900      CALL PUT(LGBSK,LGIND,CHAR)
08905      CALL PUT(SMPSK,SMIND,CHAR)
08910      RETURN
08915      END
08920      C

```

```

00025 C
00030 SUBROUTINE PUT(BSK,IND,CHAR1)
00035 C MOVES 'NEW BID' INTO PROPER ORDERED POSITION WITHIN CHARACTER TYPE
00040 C GROUPING WITHIN BASKET.
00045 C
00050 COMMON /BID/ BIDNUM(250),BIDPRI(250),BIDNEG(250),BIDTYP(250)
00055 COMMON /BASK/ BASKET(10,250),BASK1(10,150),BTOT(10),BNEG(10)
00060 COMMON /BIDORD/ BIDOR(150),NGROUP
00065 COMMON /CHRCR/ NCHAR,CHRNAM(12,2),CHRPBK(250,12)
00070 COMMON /SWAP1/LGPRK,LGIND,SMRPSK,SMIND,STARTC(12),CHAR
00075 INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR,CHRNAM,CHRPBK
00080 INTEGER LGPRK,LGIND,SMRPSK,SMIND,STARTC,CHAR
00085 INTEGER BSK,CHAR1,STRT,STP
00090 IF(BASK1(BSK,IND).EQ.-1) GO TO 4
00095 IF(IND.EQ.STARTC(CHAR1)) GO TO 2
00100 IF(BIDPRI(BASK1(BSK,IND)).LE.BIDPRI(BASK1(BSK,IND-1))) GO TO 2
00105 STRT=STARTC(CHAR1)
00110 STP=IND-1
00115 DO 1 K=STRT,STP
00120 KK=STP+STRT-K+1
00125 KK1=KK-1
00130 IF(BIDPRI(BASK1(BSK,KK)).LE.BIDPRI(BASK1(BSK,KK1))) RETURN
00135 TEMP=BASK1(BSK,KK)
00140 BASK1(BSK,KK)=BASK1(BSK,KK1)
00145 BASK1(BSK,KK1)=TEMP
00150 1 CONTINUE
00155 RETURN
00160 2 STRT=IND
00165 IF(STRT.EQ.NGROUP) RETURN
00170 STP=NGROUP-1
00175 IF(CHAR1.LT.NCHAR) STP=STARTC(CHAR1+1)-2
00180 IF(STRT.GT.STP) RETURN
00185 DO 3 K=STRT,STP
00190 IF(BASK1(BSK,K+1).EQ.-1) RETURN
00195 IF(BIDPRI(BASK1(BSK,K)).GE.BIDPRI(BASK1(BSK,K+1))) RETURN
00200 TEMP=BASK1(BSK,K)
00205 BASK1(BSK,K)=BASK1(BSK,K+1)
00210 BASK1(BSK,K+1)=TEMP
00215 3 CONTINUE
00220 RETURN
00225 4 STP=NGROUP
00230 IF(CHAR1.LT.NCHAR) STP=STARTC(CHAR1+1)-1
00235 IF(IND.EQ.STP) RETURN
00240 STP=STP-1
00245 DO 5 KK=LGIND,STP
00250 BASK1(BSK,KK)=BASK1(BSK,KK+1)
00255 BASK1(BSK,STP+1)=-1
00260 RETURN
00265 END
00270 C

```

```

00000 SUBROUTINE AVERM
00010 CALCULATED PERCENT OVER WAPL.
00020
00030 COMMON /RANGE/ NBASKL,NBASKH,NBACK
00040 COMMON /BID/ BIDNUM(150),BIDPRI(150),BIDNEG(150),BIDTYP(150)
00050 COMMON /BASK/ BASKET(10,150),BASK1(10,150),BTOT(10),BNEG(10)
00060 COMMON /BIDORD/ BIDCR(150),NGROUP
00070 COMMON /ERROR/ PCTERR,NERROR(10),ERRSUM(10),ERRSQ(10),ERRPOM(10,2)
00080 INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDCR
00090 INTEGER BSK
00100 BSK=IPAND(NBACK)
00110 BSKBID=0.
00120 BSKNEG=0.
00130 DO 1 I=1,NGROUP
00140 INDEX=BASK1(BSK,I)
00150 IF(INDEX.EQ.-1) GO TO 1
00160 BSKBID=BSKBID+BIDPRI(INDEX)
00170 BSKNEG=BSKNEG+BIDNEG(INDEX)
00180 1 CONTINUE
00190 SUMBID=0.
00200 TOTNEG=BSKNEG
00210 DECREM=BSKNEG/BSKBID
00220 DO 2 J=1,NBACK
00230 IF(J.EQ.BSK) GO TO 2
00240 DO 2 I=1,NGROUP
00250 INDEX=BASK1(J,I)
00260 IF(INDEX.EQ.-1) GO TO 2
00270 SUMBID=SUMBID+BIDPRI(INDEX)
00280 TOTNEG=TOTNEG+BIDNEG(INDEX)
00290 2 CONTINUE
00300 3 CONTINUE
00310 AWARD=BSKNEG+DECREM*SUMBID
00320 PCTERR=100.*(AWARD-TOTNEG)/TOTNEG
00330 ERRSUM(NBACK)=ERRSUM(NBACK)+PCTERR
00340 ERRSQ(NBACK)=ERRSQ(NBACK)+PCTERR**2
00350 NERROR(NBACK)=NERROR(NBACK)+1
00360 CALL FRECCY
00370 RETURN
00380 END
00390
00400 SUBROUTINE FRECCY
00410 SETS UP THE FREQ ARRAY.
00420
00430 COMMON /RANGE/ NBASKL,NBASKH,NBACK
00440 COMMON /HIST/ FREQ(21,10),BELOW(10),OUTOFF(10),OVRUNG(10,10)
00450 COMMON /ERROR/ PCTERR,NERROR(10),ERRSUM(10),ERRSQ(10),ERRPOM(10,2)
00460 INTEGER FREQ,BELOW,OUTOFF
00470 INDEX=PCTERR+11.5
00480 IF(INDEX.GT.21.OR.INDEX.LT.1) GO TO 1
00490 FREQ(INDEX,NBACK)=FREQ(INDEX,NBACK)+1
00500 RETURN
00510 1 OUTOFF(NBACK)=OUTOFF(NBACK)+1
00520 IF(INDEX.LT.1) BELOW(NBACK)=BELOW(NBACK)+1
00530 IF(OUTOFF(NBACK).LE.EQ) OVRUNG(OUTOFF(NBACK),NBACK)=PCTERR
00540 RETURN
00550 END
00560

```

```

10070 SUBROUTINE PCTMOM
10080 CALCULATES MEAN & STANDARD DEVIATION FOR PERCENT ERROR.
10090 ERMOM(NBASK,1) = MEAN FOR BASKET NBASK.
10100 ERMOM(NBASK,2) = STANDARD DEVIATION FOR BASKET NBASK.
10110
10120 COMMON /RANGE/ NBASKL,NBASKH,NBASK
10130 COMMON /ERROR/ PCTERR,NEPERR(10),ERRSUM(10),EPRSC(10),ERMOM(10,2)
10140 DO 1 NBASK=NBASKL,NBASKH
10150 ERMOM(NBASK,1)=ERRSUM(NBASK)/NEPERR(NBASK)
10160 1 ERMOM(NBASK,2)=SQRT((EPRSC(NBASK)-ERRSUM(NBASK)**2/NEPERR(NBASK))
10170 /((NEPERR(NBASK)-1)))
10180 RETURN
10190 END
10200
10210 SUBROUTINE RNSTAT
10220 OUTPUTS RUN STATISTICS.
10230
10240 COMMON /POP/ NBIDS,CHNUMP(12)
10250 COMMON /SUBPOP/ NSUBP,CHNUMS(12)
10260 COMMON /CHRCTF/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
10270 COMMON /DEPSEL/ NREP,RAND,IWEAL,NCODE
10280 INTEGER CHNUMP,CHNUMS,CHRNAM,CHRESK
10290 LOGICAL RAND
10300 WRITE(" ",1) NBIDS
10310 1 FORMAT(///,' RUN STATISTICS',//,5X,'POPULATION SIZE = ',I3,/)
10320 IF(NCHAR.EQ.1) GO TO 3
10330 WRITE(" ",2) ((CHRNAM(I,J),J=1,2),CHNUMP(I),I=1,NCHAR)
10340 2 FORMAT(10X,2A4,' = ',I3)
10350 3 WRITE(" ",4) NSUBP
10360 4 FORMAT(' ',//,5X,'SUBPOPULATION SIZE = ',I3,/)
10370 GO TO (6,11,12), NCODE
10380 5 WRITE(" ",5)
10390 6 FORMAT(/,5X,'SELECTION METHOD: SIMPLE RANDOM SAMPLING.',/)
10400 WRITE(" ",6) IWEAL
10410 7 FORMAT(5X,'NUMBER OF SELECTIONS CAPABLE OF CAUSING IWEALANCE = '
10420 2,I3)
10430 GO TO 9
10440 8 WRITE(" ",8)
10450 9 FORMAT(/,5X,'SELECTION METHOD: STRATIFIED RANDOM SAMPLING-',/,
10460 5X,'PROPORTIONAL ALLOCATION',/)
10470 GO TO 11
10480 10 WRITE(" ",10)
10490 11 FORMAT(/,5X,'SELECTION METHOD: STRATIFIED RANDOM SAMPLING-',/,
10500 5X,'USER ALLOCATION',/)
10510 12 WRITE(" ",2) ((CHRNAM(I,J),J=1,2),CHNUMS(I),I=1,NCHAR)
10520 13 WRITE(" ",10) NREP
10530 14 FORMAT(/,5X,'NUMBER OF REPLICATIONS = ',I4,/)
10540 RETURN
10550 END
10560

```

```

10480 C
10490 SUBROUTINE HISTO
10500 C
10510 COMMON /RANGE/ NRASKL,NRASKH,NRASK
10520 COMMON /REPSEL/ NREP,PAND,IMPAL
10530 COMMON /HIST/ FREQ(21,10),BELOW(10),OUTOFR(10),OVRUNS(50,10)
10540 COMMON /ERROR/ PCTERR,NERPOR(10),ERRSUM(10),ERRSO(10),ERRMOM(10,2)
10550 INTEGER FREQ,OUTOFR,BELOW
10560 LOGICAL PAND
10570 WRITE(" ",1) NRASK
10580 1 FORMAT('1-1/1-1',21X,'SIMULATION RESULTS:1/22Y,
10590 2 RELATIVE FREQUENCY OF PERCENT OVERAWARD1/24Y,'WITH',
10600 3 I2,' BASKETS')
10610 CALL HISTOGR(NREP,21,-10.,1.,FREQ(1,NRASK),BELOW(NRASK),.FALSE.)
10620 K=OUTOFR(NRASK)
10630 IF (K.EQ.0) GO TO 4
10640 WRITE(" ",2) K
10650 2 FORMAT('0',I2,' OVERAWARDS EXCEEDED THE RANGE OF THE CHART.1')
10660 IF (K.GT.50) K=50
10670 WRITE(" ",3) (OVRUNS(I,NRASK),I=1,K)
10680 3 FORMAT(10F8.2)
10690 4 WRITE(" ",5) ERRMOM(NRASK,1),ERRMOM(NRASK,2)
10700 5 FORMAT(2X,'PERCENT ERROR:1,2X,'MEAN =1,F0.3,2Y,'STD DEV =1,F0.3)
10710 WRITE(" ",6)
10720 6 FORMAT('1-1/1-1')
10730 RETURN
10740 END
10750 C
10760 SUBROUTINE HISTOGR(NTRIAL,NFREQ,CICTR,CIDEL,FREQS,BELOW,NORMLZ)
10770 C
10780 C THIS SUBROUTINE ACCEPTS AS INPUT AN INITIAL CLASS INTERVAL, A CLASS
10790 C INTERVAL DELTA, THE NUMBER OF CLASS INTERVALS FOR WHICH FREQUENCY
10800 C DATA WAS COLLECTED, THE TOTAL OF ALL FREQUENCIES IN THE EXPERIMENT
10810 C AND AN ARRAY OF FREQUENCIES, EACH CORRESPONDING TO A CLASS INTERVAL.
10820 C AS OUTPUT, IT PRODUCES A HISTOGRAM AND A LISTING OF THE RELATIVE
10830 C AND CUMULATIVE RELATIVE FREQUENCIES FOR EACH CLASS INTERVAL.
10840 C
10850 C NTRIAL - TOTAL OF ALL FREQUENCIES.
10860 C NFREQ - NUMBER OF CLASS INTERVALS.
10870 C CICTR - INITIAL CLASS INTERVAL CENTER.
10880 C CIDEL - WIDTH OF CLASS INTERVALS.
10890 C FREQS(I) - FREQUENCY CORRESPONDING TO THE ITH CLASS INTERVAL, WHERE
10900 C 'CICTR' IS THE FIRST CLASS INTERVAL.
10910 C BELOW - THE NUMBER OF SAMPLES WHICH FELL PRIOR TO THE 1ST CLASS INTER
10920 C
10930 C
10940 C INTEGER FREQS(NFREQ),PROCRT(50),PLANK/1 1/2,PLUS/1X1/2,ASTSK/1+1/
10950 C 2 ,BELOW
10960 C REAL RFREQS(50)
10970 C LOGICAL NORMLZ
10980 C

```

```

11000 PRINT HEADER FOR HISTOGRAM.
11010 IF (NORMLZ) WRITE(" ",5)
11020 1 FORMAT('0 CLASS',5X,'SUM',13X,'BAR CHART')
11030 1 INTERVAL',5X,'REL',5X,'REL',5X,'BAR CHART'
11040 1 CENTER',245X,'FREQ')
11050 1 1X,20(' '),11,50(' ')
11060 IF (.NOT.NORMLZ) WRITE(" ",5)
11070 5 FORMAT('0 CLASS',5X,'SUM',13X,'BAR CHART')
11080 1 INTERVAL',5X,'REL',5X,'REL'
11090 1 CENTER',245X,'FREQ'),7X,' 0.1 0.2 0.3 0.4 0.5 '
11100 1 ' 0.6 0.7 0.8 0.9'
11110 1 1X,20(' '),10('I----'),1(' ')
11120 XTRIAL=NTPIAL
11130 SIGREL=0.
111400 COMPUTE RELATIVE FREQUENCY FOR EACH CLASS INTERVAL, AND PLACE THE
111500 LARGEST RELATIVE FREQUENCY IN 'BIGREL'.
11160 DO 10 I=1,NFREQ
11170 RFREQS(I)=FREQS(I)/XTRIAL
11180 10 IF (RFREQS(I).GT.SIGREL) SIGREL=RFREQS(I)
111900 INITIALIZE CLASS INTERVAL CENTER AND CUMULATIVE RELATIVE FREQUENCY.
11200 CINTVL=CICTR
11210 CUMFREQ=BELOW/XTRIAL
11220 DO 30 NF=1,NFREQ
112300 CLEAR THE 50 CHARACTERS FOR THE BAR CORRESPONDING TO THIS CLASS INTERV
11240 DO 20 I=1,50
11250 20 BRCHRT(I)=BLANK
112600 'NPLUS' AND 'NPLUS' ARE THE NUMBER OF PLUSES THAT WILL BE PLACED ON
112700 THE BAR FOR THIS CLASS INTERVAL.
11280 IF (NORMLZ) XPLUS=RFREQS(NF)/SIGREL*50.
11290 IF (.NOT.NORMLZ) XPLUS=RFREQS(NF)/0.02
11300 NPLUS=XPLUS
11310 IF (NPLUS.EQ.0) GO TO 40
113200 PLACE THE PLUSES ON THE BAR.
11330 DO 30 I=1,NPLUS
11340 30 BRCHRT(I)=PLUS
113500 CHECK FOR A FRACTION OF A PLUS. IF IT EXISTS, ADD AN ASTERISK
113600 TO THE END OF THE BAR.
11370 40 IF (NPLUS.NE.XPLUS) BRCHRT(NPLUS+1)=ASTSK
113800 UPDATE THE CUMULATIVE RELATIVE FREQUENCY.
11390 CUMFREQ=CUMFREQ+RFREQS(NF)
114000 PRINT THE CLASS INTERVAL CENTER, CUMULATIVE RELATIVE FREQUENCY, RELATI
114100 FREQUENCY, AND BAR CHART FOR THIS CLASS INTERVAL.
11420 WRITE(" ",50) CINTVL,CUMFREQ,RFREQS(NF),BRCHRT
11430 50 FORMAT(F0.2,2F0.1,3X,'I',50A1)
114400 UPDATE TO THE NEXT CLASS INTERVAL CENTER.
11450 50 CINTVL=CINTVL+CIDEL
114600 PRINT THE BOTTOM BORDER AND LEAVE.
11470 IF (NORMLZ) WRITE(" ",70)
11480 11 FORMAT(1X,20(' '),11,50(' '))
11490 IF (.NOT.NORMLZ) WRITE(" ",50)
11500 11 FORMAT(1X,20(' '),10('I----'),1(' '))
11510 RETURN
11520 END
115300

```



```

11500      SUBROUTINE RETAPL (NREQ,NREPS)
11505      COMMON /RANGE/ NRASKL,NRASKH,NRASK
11510      COMMON /HIST/ FREQ(21,10),IPLANK(20),BLANK(50,10)
11515      INTEGER FREQ
11520      REAL RFREQ(21,10),CFREQ(6)
11525      WRITE("      ",101)
11530      WRITE("      ",102)
11535      WRITE("      ",103)
11540      WRITE("      ",104)
11545      WRITE("      ",105)
11550      XNREPS=NREPS
11555      DO 10 NRSK=NRASKL,NRASKH
11560      DO 10 NF=1,NREQ
11565      10 RFREQ(NF,NRSK)=FREQ(NF,NRSK)/XNREPS
11570      DO 30 NRSK=NRASKL,NRASKH
11575      MIDDLE=NREQ/2+1
11580      CTOT=RFREQ(MIDDLE,NRSK)
11585      DO 20 I=1,6
11590      CFREQ(I)=CTOT+RFREQ(MIDDLE-I,NRSK)+RFREQ(MIDDLE+I,NRSK)
11595      20 CTOT=CFREQ(I)
11600      WRITE("      ",106)
11605      WRITE("      ",107)NRSK,(CFREQ(I),I=1,6)
11610      WRITE("      ",106)
11615      30 WRITE("      ",105)
11620      101 FORMAT(' ',25X,'RELATIVE FREQUENCY OF PERCENT OVERAWARD')
11625      102 FORMAT(' ',11X,'NUMBER',10X,'PERCENT OVERAWARD INTERVAL')
11630      103 FORMAT(' ',13X,'OF')
11635      104 FORMAT(' ',11X,'ASKTS',5X,'+-1',7X,'+-2',7X,'+-3',7X,'+-4',7X,
11640      *      '+-5',7X,'+-6')
11645      105 FORMAT(' ',10X,69(' '))
11650      106 FORMAT(' ',10X,' ',6X,' ',6(9X,' '))
11655      107 FORMAT(' ',10X,' ',2X,I2,2X,' ',6(2X,F5.3,2X,' '))
11660      RETURN
11665      END
11670 C

```

```

11675 C
11680 SUBROUTINE PRICE(*)
11685 COMMON /PERSEL/ NREP, RAND, IMBAL, NCODE
11690 LOGICAL RAND
11695 CALL PRINPU
11700 CALL PRINT
11705 IF(RAND) GO TO 1
11710 CALL CHOICE($2)
11715 1 CALL RANDOM
11720 2 CALL PDORD
11725 CALL BASKT
11730 CALL SWAP
11735 CALL NEGTOT
11740 CALL LISTB($3)
11745 CALL STATS
11750 CALL OUTPUT
11755 3 CALL LISTP($4)
11760 CALL PRICOM
11765 CALL LSTPRC
11770 GO TO 3
11775 4 RETURN
11780 END
11785 C
11790 C
11795 SUBROUTINE PRINPU
11800 CALL NUMPSK
11805 CALL NUMSUR
11810 CALL TYPSEL
11815 RETURN
11820 END
11825 C
11830 C
11835 SUBROUTINE NUMPSK
11840 C INPUTS NUMBER OF BASKETS INTO - NBASK.
11845 C
11850 COMMON /RANGE/ NBASKL, NBASKH, NBASK
11855 COMMON /POP/ NIDS, CHNIMP(12)
11860 COMMON /SUBPOP/ NSUPP, CHNUMS(12)
11865 INTEGER CHNIMP, CHNUMS
11870 1 WRITE(" ", 2)
11875 2 FORMAT(' ENTER NUMBER OF BASKETS. ')
11880 READ(" ", *) NBASK
11885 NBASKH=NBASK
11890 IF(2.LE.NBASK.AND.NBASK.LE.10) RETURN
11895 WRITE(" ", 3) NBASK
11900 3 FORMAT(/, ' *** ERROR *** NUMBER OF BASKETS SPECIFIED, '14,
11905 2 15X, 'IS NOT BETWEEN 2 AND 10 INCLUSIVE. ', /, ' DO YOU WISH',
11910 2 15X, ' TO REENTER NUMBER OF BASKETS? (YES OR NO) ')
11915 CALL ANSWER($1)
11920 STOP
11925 END
11930 C
11935 C

```

```

121010 SUBROUTINE NESTOT
121020
121030 CALCULATES TOTALS OF NEGOTIATED PRICES FOR EACH BASKET
121040 AND PLACES INTO = ENEG(I).
121050
121060 COMMON /RANGE/ NBASKL,NBASKH,NBASK
121070 COMMON /EID/ EIDNUM(350),EIDPRI(350),EIDNEG(350),EIDTYP(350)
121080 COMMON /BACK/ BASKET(10,350),BASK1(10,160),BTOT(10),ENEG(10)
121090 COMMON /BIDORD/ BIDOR(160),NGROUP
121100 INTEGER EIDNUM,EIDTYP,BASKET,BASK1,BIDOR
121110 DO 1 I=1,NBASK
121120 ENEG(I)=0.
121130 DO 1 J=1,NGROUP
121140 IF(BASK1(I,J).EQ.-1) GO TO 1
121150 ENEG(I)=ENEG(I)+BIDNEG(BASK1(I,J))
121160 1 CONTINUE
121170 RETURN
121180 END
122100
122110 SUBROUTINE LISTP(*)
122120 LOGICAL ARG
122130 WRITE(" ",1)
122140 PRINT,""
122150 1 FORMAT(/,' DO YOU WISH TO LIST BASKETS AND BASKET STATISTICS?'
122160 &,' (YES OR NO)')
122170 CALL ANSWR1(ARG)
122180 IF(ARG) RETURN
122190 RETURN 1
122200 END
123100
123110 SUBROUTINE STATE
123120 CALL PCTERS
123130 CALL MCNMTS
123140 RETURN
123150 END
124100
124110 SUBROUTINE PCTERE
124120 CALCULATES PER CENT ERROR BOUNDS.
124130
124140 COMMON /RANGE/ NBASKL,NBASKH,NBASK
124150 COMMON /BACK/ BASKET(10,350),BASK1(10,160),BTOT(10),ENEG(10)
124160 COMMON /EFFEND/ MAXDIF,LOWEND,UPPBRND,BOKEST(10),ESTDIF(10),
124170 PCTEST(10)
124180 INTEGER BASKET,BASK1
124190 INTEGER BOK,ORDER1(10)
124200 REAL LOWEND,MAX,MIN,MAXDIF
124210 CALL ORDER(10,ORDER1,BTOT,NBASK)
124220 MAX=BTOT(ORDER1(NBASK))
124230 MIN=BTOT(ORDER1(1))
124240 MAXDIF=MAX-MIN
124250 TOTL=0.
124260 TOTNEG=0.

```

```

12550      DO 1 BSK=1, NBASK
12560          TCTNEG=TCTNEG+BNEG(BSK)
12570          TOTAL=TOTAL+BTOT(BSK)
12580      1      LEVEND=100.*(TOTAL/NBASK/NAX-1.)
12590          UPREND=100.*(TOTAL/NBASK/NTM-1.)
12600          DO 2 BSK=1, NBASK
12610              DECREM=BNEG(BSK)/BTOT(BSK)
12620              DECBID=DECREM*(TOTAL-BTOT(BSK))
12630              BSKEST(BSK)=DECBID+BNEG(BSK)
12640              ESTDIF(BSK)=DECBID-TCTNEG+BNEG(BSK)
12650      2      PCTEST(BSK)=ESTDIF(BSK)/TCTNEG*100.
12660          RETURN
12670      END

```

```

12680C
12690C
12700C
12710C
12720C
12730C

```

SUBROUTINE MCNMTS  
CALCULATES MEAN, STD DEV FOR EACH BASKET AND POPULATION.

```

12740      COMMON /RANGE/ NBASKL, NBASKH, NBASK
12750      COMMON /SUBPOP/ NSUBP, CHNUMS(12)
12760      COMMON /BID/ BIDNUM(350), BIDPRI(350), BIDNEG(350), BIDTYP(350)
12770      COMMON /BASK/ BASKET(10, 350), BASK1(10, 150), BTOT(10), BNEG(10)
12780      COMMON /BIDORD/ BIDOR(150), NGROUP
12790      COMMON /MCN/ BCKVAR(10), BSKVAR(10), POPVM, POPVAR, NBASK1(10), PCTOT
12800      , POPNEG
12810      INTEGER CHNUMS, BIDNUM, BIDTYP, BASKET, BASK1, BIDOR
12820      INTEGER BSK
12830      DO 1 BSK=1, NBASK
12840      1      BSKVAR(BSK)=0.
12850          POPVM=0.
12860          POPVAR=0.
12870          POPNEG=0.
12880          DO 3 BSK=1, NBASK
12890              NBID=0
12900              POPNEG=POPNEG+BNEG(BSK)
12910              DO 2 N=1, NGROUP
12920                  INDEX=BASK1(BSK, N)
12930                  IF (INDEX.EQ.-1) GO TO 2
12940                  NBID=NBID+1
12950                  BSKVAR(BSK)=BSKVAR(BSK)+BIDPRI(INDEX)**2
12960      2      CONTINUE
12970          POPVM=POPVM+BTOT(BSK)
12980          POPVAR=POPVAR+BSKVAR(BSK)
12990          TSKVM(BSK)=BTOT(BSK)/NBID
13000          BSKVAR(BSK)=SQRT((BSKVAR(BSK)-BTOT(BSK)**2/NBID)/(NBID-1))
13010          NBASK1(BSK)=NBID
13020      3      CONTINUE
13030          POPVAR=SQRT((POPVAR-POPVM**2/NSUBP)/(NSUBP-1))
13040          PCTOT=POPVM
13050          POPVM=POPVM/NSUBP
13060          RETURN
13070      END
13080C

```

```

12100C      SUBROUTINE OUTPUT
12110C      CALL RSTATS
12120C      CALL LSTRESK
12130C      CALL RSTATS
12141C      RETURN
12150C      END
12160C
12170C
12180C      SUBROUTINE RSTATS
12190C      PRINTS RUN STATISTICS.
12200C
12210C      COMMON /POP/ NIDS,CHNUP(12)
12220C      COMMON /SUBPOP/ NSUPP,CHNUMS(12)
12230C      COMMON /CHACTR/ NCHAR,CHENAV(12,2),CHRESK(150,12)
12240C      COMMON /REPSL/ NREP,RAND,IMBAL,NCODE
12250C      INTEGER CHNUP,CHNUMS,CHENAV,CHRESK
12260C      LOGICAL RAND
12270C      WRITE(" ",1) NIDS
12280C      1 FORMAT(///,' RUN STATISTICS',//,5X,'POPULATION SIZE = ',I3,/)
12290C      IF(NCHAR.EQ.1) GO TO 3
12300C      WRITE(" ",2) ((CHENAV(I,J),J=1,2),CHNUP(I),I=1,NCHAR)
12310C      2 FORMAT(10X,2A4,' = ',I3)
12320C      3 WRITE(" ",4) NSUPP
12330C      4 FORMAT(' ',//,5X,'SUBPOPULATION SIZE = ',I3,/)
12340C      GO TO (6,8,9), NCODE
12350C      6 WRITE(" ",7)
12360C      7 FORMAT(/,5X,'SELECTION METHOD:  SIMPLE RANDOM SAMPLING.',/)
12370C      RETURN
12380C      8 WRITE(" ",5)
12390C      5 FORMAT(/,5X,'SELECTION METHOD:  STRATIFIED RANDOM SAMPLING-',/,
12400C      127X,'PROPORTION ALLOCATION',/)
12410C      GO TO 12
12420C      9 WRITE(" ",10)
12430C      10 FORMAT(/,5X,'SELECTION METHOD:  STRATIFIED RANDOM SAMPLING-',/,
12440C      127X,'USER ALLOCATION',/)
12450C      12 WRITE(" ",2) ((CHENAV(I,J),J=1,2),CHNUMS(I),I=1,NCHAR)
12460C      RETURN
12470C      END
12480C

```

124000  
 124100  
 124200  
 124300  
 124400  
 124500  
 124600  
 124700  
 124800  
 124900  
 125000  
 125100  
 125200  
 125300  
 125400  
 125500  
 125600  
 125700  
 125800  
 125900  
 126000  
 126100  
 126200  
 126300  
 126400  
 126500  
 126600  
 126700  
 126800  
 126900  
 127000  
 127100  
 127200  
 127300  
 127400  
 127500  
 127600  
 127700  
 127800  
 127900  
 128000  
 128100  
 128200  
 128300  
 128400  
 128500  
 128600  
 128700  
 128800  
 128900  
 129000

SUBROUTINE LINTASK  
 LINTO BASKETS.

COMMON /RANGE/ NRASKL,NRASKH,NRASK  
 COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)  
 COMMON /BASK/ BASKT(10,350),BASK1(10,100),BCTOT(10),BNEG(10)  
 COMMON /BIDGRP/ BIDGR(100),NGRGRP  
 COMMON /CHRCR/ NCHAR,CHRNAM(12,2),CHRBRSK(350,12)  
 INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDGR,CHRNAM,CHRBRSK  
 INTEGER BSK  
 DO 10 BSK=1,NRASK  
 WRITE(" ",1) BSK  
 1 FORMAT(' ',//,5X,'BASKET NO.',5X,10)  
 IF(NCHAR.GT.1) GO TO 2  
 WRITE(" ",2)  
 2 FORMAT(' ',//,12X,'PROPOSAL NO.',10X,'BID PRICE',5X,'NEGOTIATED ',  
 'PRICE',/)  
 GO TO 3  
 3 WRITE(" ",4)  
 4 FORMAT(' ',12X,'PROPOSAL NO.',10X,'BID PRICE',5X,'NEGOTIATED ',  
 'PRICE',2X,'CHARACTERISTIC',/)  
 5 MINUS=0  
 DO 6 N=1,NGRGRP  
 NN=N-MINUS  
 INDEX=BASK1(BSK,N)  
 IF(INDEX.EQ.-1) GO TO 6  
 IF(NCHAR.EQ.1) GO TO 7  
 WRITE(" ",6)NN,INDEX,BIDPRI(INDEX),BIDNEG(INDEX),  
 1(CHRNAM(BIDTYP(INDEX),1),I=1,2)  
 6 FORMAT(T10,I3,' ',T17,I4,T35,F10.3,T50,F10.3,T70,2(4)  
 GO TO 6  
 7 MINUS=MINUS+1  
 GO TO 6  
 8 WRITE(" ",8) N,INDEX,BIDPRI(INDEX),BIDNEG(INDEX)  
 9 CONTINUE  
 10 CONTINUE  
 11 WRITE(" ",11)  
 11 FORMAT(' ',5X,40(' '))  
 RETURN  
 END

```

141130
141140
141150
141160
141170
141180
141190
141200
141210
141220
141230
141240
141250
141260
141270
141280
141290
141300
141310
141320
141330
141340
141350
141360
141370
141380
141390
141400
141410
141420
141430
141440
141450
141460
141470
141480
141490
141500
141510
141520
141530
141540
141550
141560
141570
141580
141590
141600
141610
141620
141630
141640
141650
141660
141670
141680
141690
141700
141710
141720
141730
141740
141750
141760
141770
141780
141790
141800
141810
141820
141830
141840
141850
141860
141870
141880
141890
141900
141910
141920
141930
141940
141950
141960
141970
141980
141990
142000

      TITLE 'BID PRICE MOMENTS'
      PRINTS 'BID PRICE STATISTICS'

      COMMON /SUBPOP/ NSUBP,CHUNKS(12)
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /BASK/ BASKET(10,350),BASK1(10,10),BTOT(10),BNEG(10)
      COMMON /ERRRND/ MAXDIF,LOWEND,UPREND,BSEKST(10),ESTDIF(10),
141200 PTEST(10)
      COMMON /VCM/ BSKWH(10),BSKVAR(10),PCPMN,POPVAR,NBASK1(10),POPTOT
141210 ,POPNEG
141220
141230 INTEGER CHUNKS,BASKET,BASK1
141240 REAL LOWEND,MAX,MIN,MAXDIF
141250 WRITE(" ",1)
141260 1 FORMAT(' ',/2X,'BASKET STATISTICS',/)
141270 WRITE(" ",2) MAXDIF
141280 2 FORMAT(T3,'DIFFERENCE BETWEEN MAX AND MIN BASKETS =',2X,F10.2)
141290 WRITE(" ",3) LOWEND,UPREND
141300 3 FORMAT(T3,'MEAN ERROR BOUNDED BY ',F5.2,'% AND ',F5.2,'%')
141310 WRITE(" ",4)
141320 4 FORMAT(T3,'BID PRICE MOMENTS:')
141330 WRITE(" ",5)
141340 5 FORMAT(/,2X,'NO.',5X,'TOTAL',7X,'TOTAL',20X,'POPUL',5X,'PERCENT',
141350 1X,'BASKET',2X,'BIDS',4X,'OF BIDS',5X,'NEGOT.',7X,'MEAN',
141360 5X,'STD DEV',5X,'VARIANCE',4X,'ERROR',/)
141370 DO I=1,NBASK
141380 6 WRITE(" ",7) I,NBASK1(I),BTOT(I),BNEG(I),BSKWH(I),BSKVAR(I),
141390 BSEKST(I),PTEST(I)
141400 7 FORMAT(2X,I2,4X,I3,5X,F10.2,2X,F10.2,2X,F5.2,2X,F5.2,2X,F10.2,1X,
141410 F5.2)
141420 WRITE(" ",8) NSUBP,POPTOT,POPNEG,PCPMN,POPVAR
141430 8 FORMAT(/,1X,'POPUL',2X,I3,3X,F10.2,2X,F10.2,2X,F5.2,2X,
141440 F5.2,/,2X,50(' '),/)
141450 RETURN
141460 END
141470 SUBROUTINE LISTP(*)
141480 COMMON /RANGE/ NBASKL,NBASKH,NBASK
141490 COMMON /PRICES/ BSKSEL,ESTIN(350),ERROR(350),PCTERR(350)
141500 ,ISPCHR(350),TOTBID,TOTNEG,TOTEST,TOTERR,MEAN
141510
141520 INTEGER BSKSEL
141530 LOGICAL ARG
141540 WRITE(" ",1)
141550 PRINT,""
141560 1 FORMAT(/,' DO YOU WISH TO MAKE A BASKET SELECTION FOR PRICING?',
141570 1X,' (YES OR NO)')
141580 CALL ANSWR1(ARG)
141590 IF(.NOT.ARG) RETURN
141600 2 WRITE(" ",2) NBASK
141610 PRINT,""
141620 3 FORMAT(' ENTER BASKET NUMBER BETWEEN 1 AND ',I2,' INCLUSIVE.')
141630 READ(" ",*) BSKSEL
141640 IF(1.LE.BSKSEL.AND.BSKSEL.LE.NBASK) RETURN
141650 WRITE(" ",4) BSKSEL
141660 4 FORMAT(/,' *** ERROR *** REQUESTED BASKET, ',I2,' IS NOT VALID.')
141670 1X,15X,' DO YOU WISH TO REENTER? (YES OR NO)'
141680 CALL ANSWR(2)
141690 STOP
141700 END
141710
141720
141730
141740
141750
141760
141770
141780
141790
141800
141810
141820
141830
141840
141850
141860
141870
141880
141890
141900
141910
141920
141930
141940
141950
141960
141970
141980
141990
142000

```

```

14510 SUBROUTINE PRICOM
14520 CALCULATED PRICES WHEN DECREMENT IS APPLIED TO OTHER BASKETS.
14530 COMMON /PCP/ NEIDC,CHUNKP(12)
14540 COMMON /RANGE/ NBASKL,NBASKH,NBASK
14550 COMMON /BID/ BIDNUM(350),BIDPRI(150),BIDNEG(350),BIDTYP(350)
14560 COMMON /BACK/ BASKET(10,350),BASK1(10,100),BTOT(10),BNEG(10)
14570 COMMON /BIDORD/ BIDOR(100),NGROUP
14580 COMMON /PRICES/ PCKSEL,ESTIM(350),ERROR(350),PCTERR(350)
14590 ,ISPCNR(350),TOTRID,TOTNEG,TOTEST,TOTERR,MEAN
14600 COMMON /TAB1/ TAB(350)
14610 LOGICAL TAB
14620 INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR,BCKSEL
14630 REAL MEAN
14640 INTEGER I,ISTK,BLANK
14650 DATA ISTK/'#'/,BLANK/' '/
14660 DECREM=BNEG(BCKSEL)/BTOT(BCKSEL)
14670 TOTRID=0.
14680 TOTNEG=0.
14690 TOTEST=0.
14700 TOTERR=0.
14710 DO 10 I=1,NEIDC
14720   TAB(I)=.FALSE.
14730   DO 4 J=1,NBASKH
14740     TOTRID=TOTRID+BIDTYP(I)
14750     TOTNEG=TOTNEG+BNEG(I)
14760     IF(I.EQ.BCKSEL) GO TO 3
14770     DO 1 J=1,NGROUP
14780       IF(BASK1(I,J).EQ.-1) GO TO 1
14790       INDEX=BASK1(I,J)
14800       TAB(INDEX)=.TRUE.
14810       ESTIM(INDEX)=DECREM*BIDPRI(INDEX)
14820       ISPCNR(INDEX)=BLANK
14830       ERROR(INDEX)=ESTIM(INDEX)-BIDNEG(INDEX)
14840       PCTERR(INDEX)=ERROR(INDEX)/BIDNEG(INDEX)*100.
14850       TOTERR=TOTERR+ERROR(INDEX)
14860       TOTEST=TOTEST+ESTIM(INDEX)
14870     1 CONTINUE
14880     GO TO 4
14890   2 DO 3 J=1,NGROUP
14900     IF(BASK1(I,J).EQ.-1) GO TO 3
14910     INDEX=BASK1(I,J)
14920     TAB(INDEX)=.TRUE.
14930     ESTIM(INDEX)=BIDNEG(INDEX)
14940     ISPCNR(INDEX)=ISTK
14950     ERROR(INDEX)=ESTIM(INDEX)-BIDNEG(INDEX)
14960     PCTERR(INDEX)=ERROR(INDEX)/BIDNEG(INDEX)*100.
14970     TOTERR=TOTERR+ERROR(INDEX)
14980     TOTEST=TOTEST+ESTIM(INDEX)
14990   3 CONTINUE
15000   4 CONTINUE
15010   MEAN=TOTERR/TOTNEG*100.
15020   RETURN
15030   END
15040

```



```

15041      NEGATIVE LUTPG
15050      LIST PRICES.
15061      COMMON /BID/ NRIDS,CHUNK(12)
15071      COMMON /CHRPOR/ NCHRP,CHUNK(12)
15080      COMMON /BID/ BIDNUM(150),BIDPRI(150),BIDNEG(150),BIDTYP(150)
15090      COMMON /PRICES/ BSKSEL,ESTIM(150),ERRPR(150),PCTERR(150)
15100      ,ISPCNR(150),TOTFID,TOTNEG,TOTEST,TOTERR,MEAN
15110      COMMON /TAB1/ TAB(150)
15120      LOGICAL TAB
15130      INTEGER CHUNKS,BKSEL,BIDNUM,BIDTYP
15140      REAL MEAN
15150      WRITE(" ",1) BSKSEL
15160      1  FORMAT(//,' BASKET ',12,' SELECTED.')
15170      WRITE(" ",2)
15180      2  FORMAT(/,' PROPOSAL',10X,'BID',5X,'NEGOTIATED',5X,'ESTIMATED'
15190      ',5X,'ERROR',5X,'PERCENT'/' NUMBER',3(10X,'PRICE'),10X,'ERPOP'
15200      '/')
15210      DO 4 I=1,NRIDS
15220      IF(.NOT.TAB(I)) GO TO 4
15230      WRITE(" ",3) I,BIDPRI(I),BIDNEG(I)
15240      ,ESTIM(I),ERRPR(I),ISPCNR(I),PCTERR(I)
15250      3  FORMAT(I9,3F15.3,3F12.3,A1,F11.2)
15260      4  CONTINUE
15270      WRITE(" ",5) TOTFID,TOTNEG,TOTEST,TOTERR,MEAN
15280      5  FORMAT(10X,3(5X,2(' ')),5X,7(' ')/' TOTALS',3F15.3,F12.3/
15290      ' PERCENT ERROR IS',F6.2,'%')
15300      RETURN
15310      END

```

BALSEL 13:PM PDT 07/10/81

1 DESCRIPTION PRTP

1000 BASKET METHOD

1100

1200 ALGORITHM DESIGN AND IMPLEMENTATION - DR. STEPHEN L. BENZ  
1300 GLENSON UNIVERSITY  
1400 GLENSON, S.C. 29631

1500

1600 VERSION III, JUNE 1981.

1700

1800 THIS ALGORITHM SOLVES THE FOLLOWING PROBLEM.

1900

2000 GIVEN: (1) A SET OF BIDPRICE DATA WITH EACH ITEM TYPED BY  
2100 SOME OUTSIDE CRITERION (1 TO 12 TYPES).

2200

2300

2400

2500

2600

2700

2800

2900

3000

3100

3200

3300

3400

3500

3600

3700

3800

3900

4000

4100

4200

4300

4400

4500

4600

4700

4800

4900

5000

5100

5200

5300

5400

5500

5600

5700

- (2) NUMBER OF DESIRED BASKETS (2 TO 10 BASKETS).

PLACE THE BIDS INTO THE BASKETS TO OBTAIN THE FOLLOWING:

- (1) THE BIDS IN EACH BASKET SHOULD BE REPRESENTATIVE IN SIZE OF THE PARENT POPULATION.
- (2) THE COLLECTIVE BIDS OF ANY TYPE WITHIN EACH BASKET SHOULD BE REPRESENTATIVE OF THE PARENT TYPE POPULATION.
- (3) THE TOTALS OF THE BASKETS SHOULD BE EQUAL.
- (4) THE NUMBER OF BIDS IN ANY BASKET SHOULD BE WITHIN ONE OF THE NUMBER IN ANY OTHER BASKET.
- (5) THE NUMBER OF BIDS OF ANY TYPE IN ANY BASKET SHOULD BE WITHIN ONE OF THE NUMBER OF BIDS OF THE SAME TYPE IN ANY OTHER BASKET.

THE ALGORITHM IS AS FOLLOWS.

NCHAR - THE NUMBER OF CHARACTER TYPES.

NBASK - THE NUMBER OF BASKETS DESIRED.

- (1) THE BIDS ARE PLACED INTO NCHAR BASKETS BY CHARACTER TYPES AND ORDERED WITHIN EACH BASKET BY SIZE FROM LARGEST TO SMALLEST.
- (2) THE BIDS ARE PLACED FROM THESE NCHAR BASKETS INTO THE GIVEN NBASK BASKETS BY GROUPS OF NBASK, ALL OF A SINGLE TYPE. THE TYPE TO BE PLACED IS DETERMINED TO BE THE ONE HAVING THE LARGEST AVERAGE OF THE NEXT NBASK BIDS. THESE BIDS ARE THEN PLACED INTO THE BASKETS, THE SMALLEST BID GOING TO THE BASKET WITH THE LARGEST TOTAL, THE NEXT SMALLEST BID GOING TO THE BASKET WITH THE NEXT LARGEST TOTAL, AND SO ON UNTIL ALL THE BIDS OF THAT GROUPING HAVE BEEN PLACED. THIS PROCESS IS REPEATED UNTIL ALL THE BIDS HAVE BEEN PLACED.
- (3) SINCE THE ABOVE PLACEMENT OF THE BIDS DOES NOT GUARANTEE EQUAL BASKET TOTALS, BIDS OF SIMILAR TYPE ARE SWAPPED BETWEEN THE LARGEST AND THE SMALLEST BASKETS UNTIL SWAPPING WILL NO LONGER IMPROVE THE BALANCE OF THE BASKET TOTALS.

```

580C
590C      MAIN PROGRAM
600      CALL DSPLAY
610      CALL INPUT
620      1  CALL CHBASK
630      CALL BSKORD
640      CALL BIDORD
650      CALL BASKT
660      CALL SWAP
670      2  CALL STATS
680      CALL OUTPUT
690      CALL CONT(31,$2)
700      STOP
710      END
720C
730C
740      SUBROUTINE DSPLAY
750      WRITE("      ",1)
760      1  FORMAT(' ',76('*'))/' ',74X,'*'
770      &/' ',30X,'BASKET METHOD',31X,'*'/
780      &' ',24X,'VERSION III - JUNE 1981',27X,'*'/ ' ',74X
790      &,'*'/ ' ',14X,'THEORETICAL DEVELOPMENT - DR. K. T. WALLENIUS',
800      &15X,'*'/ ' ',10X,'ALGORITHM DESIGN AND IMPLEMENTATION -',
810      &' DR. STEPHEN BENZ',10X,'*'/ ' ',74X,'*'/ ' ',28X,'CLEMSON',
820      &' UNIVERSITY',28X,'*'/ ' ',31X,'CLEMSON S.C.',31X,'*'/ ' ',
830      &74X,'*'/ ' ',76('*'))
840      RETURN
850      END
860C
870      SUBROUTINE INPUT
880      INTEGER CHRNAM,CHRNUM,CHRESK
890      COMMON /CHRCR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRESK(12,300)
900      CALL NUMESK
910      CALL NUMCHR
920      IF(NCHAR.EQ.1) GO TO 1
930      CALL CHRTYP
940      1  CALL BIDS
950      RETURN
960      END
970C

```

```

9000
990      SUBROUTINE NUMBSK
1000C    READS NUMBER OF BASKETS INTO - NBASK.
1010C
1020      INTEGER ANSWR,YES,NO
1030      INTEGER BASKET,BASK1
1040      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTCT(10),NGROUP
1050      DATA YES/'Y'//,NO/'N'//
1060      1  WRITE("      ",2)
1061 PRINT,""
1070      2  FORMAT(' ENTER NUMBER OF BASKETS.')
1080      READ("      ",*) NBASK
1090      IF(NBASK.GE.2.AND.NBASK.LE.10) RETURN
1100      WRITE("      ",3) NBASK
1110      3  FORMAT(' ',/, ' *** ERROR ***  NUMBER OF BASKETS SPECIFIED,',I4,', '
1120      ',/,15X,' IS NOT BETWEEN 2 AND 10 INCLUSIVE.')
1130      WRITE("      ",4)
1131 PRINT,""
1140      4  FORMAT(' DO YOU WISH TO REENTER NUMBER OF BASKETS? (YES OR NO)')
1150      5  READ("      ",6) ANSWR
1160      6  FORMAT(A1)
1170      IF(ANSWR.EQ.YES) GO TO 1
1180      IF(ANSWR.EQ.NO) STOP
1190      WRITE("      ",7)
1191 PRINT,""
1200      7  FORMAT('REENTER "YES OR NO"')
1210      GO TO 5
1220      END
1230C
1240C
1250      SUBROUTINE NUMCHR
1260C    READS NUMBER OF CHARACTER TYPES INTO - NCHAR.
1270C
1280      INTEGER ANSWR,YES,NO
1290      INTEGER CHRNAM,CHNUM,CHRBSK
1300      COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRBSK(12,300)
1310      DATA YES/'Y'//,NO/'N'//
1320      1  WRITE("      ",2)
1321 PRINT,""
1330      2  FORMAT(' ',/, ' ENTER NUMBER OF CHARACTER TYPES.')
1340      READ("      ",*) NCHAR

```

```

1505      IF (N.GT.1.AND.NCHAR.LE.12) RETURN
1510      WRITE (" ",2) NCHAR
1520      1  FORMAT(' ',4,' *** ERROR *** NUMBER OF CHARACTER TYPES ',I3,' ',
1530      ' ',17X,' IS NOT BETWEEN 1 AND 12 INCLUSIVE.')
1540      WRITE (" ",3)
1550      PRINT,""
1560      4  FORMAT(' DO YOU WISH TO REENTER NUMBER OF CHARACTER TYPES?',
1570      ' (YES OR NO)')
1580      5  READ (" ",4) ANSWR
1590      6  FORMAT(A1)
1600      IF(ANSWR.EQ.YES) GO TO 1
1610      IF(ANSWR.EQ.NO) STOP
1620      WRITE (" ",3)
1630      PRINT,""
1640      7  FORMAT('REENTER "YES OR NO"')
1650      GO TO 5
1660      END
1670C
1680C
1690C      SUBROUTINE CHRTYP
1700C      READS CHARACTER TYPES INTO - CHRNAM(N,K), K=1,2.
1710C
1720C      INTEGER ANSWR,YES,BLANK,NO
1730C      INTEGER CHRNAM,CHRNUM,CHRSK
1740C      COMMON /CHRTYP/ NCHAR,CHRNAM(12,2),CHRTUN(12),CHRSK(12,300)
1750C      DATA YES/'Y'/,BLANK/' '/,NO/'N'/
1760      1  WRITE (" ",2) NCHAR
1770      PRINT,""
1780      2  FORMAT(' ',/, ' ENTER ',I3,' ', 3 - CHARACTER ID'S FOR CHARACTER',
1790      ' TYPES, ONE PER LINE.')
1800      DO 3 J=1,NCHAR
1810      3  READ (" ",4) (CHRNAM(J,K),K=1,2)
1820      4  FORMAT(2A4)
1830      DO 5 N=1,NCHAR
1840      IF(CHRNAM(N,1).EQ.BLANK.AND.CHRNAM(N,2).EQ.BLANK) GO TO 6
1850      5  CONTINUE
1860      RETURN
1870      6  WRITE (" ",7) NCHAR
1880      7  FORMAT(' ',/, ' *** ERROR *** NUMBER OF CHARACTER TYPES DOES ',/,
1890      ' 217X, ' NOT EQUAL ',I3,' ', OR DATA NOT ENTERED IN PROPER FORMAT.')
1900      WRITE (" ",8)
1910      PRINT,""
1920      8  FORMAT(' DO YOU WISH TO REENTER CHARACTER TYPES? (YES OR NO)')
1930      9  READ (" ",10) ANSWR
1940      10  FORMAT(A1)
1950      IF(ANSWR.EQ.YES) GO TO 1
1960      IF(ANSWR.EQ.NO) STOP
1970      WRITE (" ",11)
1980      PRINT,""
1990      11  FORMAT('REENTER "YES OR NO"')
2000      GO TO 3
2010      END
2020C

```

```

1850C
1860 SUBROUTINE BIDS
1870C READS DATA.
1880C BIDNUM(N) - BID PROPOSAL NUMBER.
1890C BIDPRI(N) - BID PRICE.
1900C BIDTYP(N) - INTEGER VALUES FROM 1 TO 12 CORRESPONDING
1910C TO TYPE OF BID.
1920 INTEGER EDTYP1,BDTYP2,ANSWR,YES,NO
1930 STRING FILENAM
1940 INTEGER BIDNUM,BIDTYP,BIDOR
1950 COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
1960C INTEGER CHRNUM,CHRESK,PROCP
1970C COMMON /CHRCTR/ NCHAR,CHRNUM(12,2),CHRESK(12,300)
1980C DATA YES/'Y'//,NO/'N'//
1990 101 PRINT,"ENTER FILENAME"
2000 INPUT, FILENAM
2010 OPEN(FILENAM,STATUS="OLD",UNIT=9,ERR=110)
2020 GOTO 100
2030 110 PRINT,"FILE NOT FOUND"
2040 GOTO 101
2050 100 NERR=0
2060 IF(NCHAR.GT.1) GO TO 2
2070 DO 1 NBIDS=1,300
2080 BIDNUM(NBIDS)=NBIDS
2090 1 READ(9,*,END=11) BIDPRI(NBIDS)
2100 GO TO 10
2110 2 DO 9 NBIDS=1,300
2120 BIDNUM(NBIDS)=NBIDS
2130 3 READ(9,*,END=11) BIDPRI(NBIDS),BIDTYP(NBIDS)
2140 IF(1.LE.BIDTYP(NBIDS).AND.BIDTYP(NBIDS).LE.NCHAR) GO TO 9
2150 PROCP=BIDNUM(NBIDS)+NERR
2160 WRITE(" ",4) PROCP,BIDTYP(NBIDS)
2170 4 FORMAT(' ',/, ' *** ERROR ENCOUNTERED ON PROPOSAL ',I4, '.'/
2180 &' THE TYPE OF PROPOSAL ',I3, ' DOES NOT MATCH WITH ANY OF ',
2190 &' THOSE SUBMITTED.')
2200 NERR=NERR+1
2210 WRITE(" ",5)
2220 PRINT,""
2230 5 FORMAT(' DO YOU WISH TO OMIT THIS PROPOSAL FROM THE ANALYSIS ',
2240 &' AND CONTINUE? (YES OR NO)')
2250 6 READ(" ",7) ANSWR
2260 7 FORMAT(A1)
2270 IF(ANSWR.EQ.YES) GO TO 3
2280 IF(ANSWR.EQ.NO) STOP
2290 WRITE(" ",8)
2300 PRINT,""
2310 8 FORMAT('REENTER "YES OR NO"')
2320 GO TO 6
2330 9 CONTINUE
2340 10 NBIDS=301
2350 11 NBIDS=NBIDS-1
2360 RETURN
2370 END
2380C
2390C

```

```

22900
23000
2310      SUBROUTINE CHEASK
23200      PUTS BIDS INTO BASKETS - CHRBSK(N,K) - BY TYPE.
23300      CHRNUM(N) - NUMBER OF BIDS OF CHARACTER TYPE N.
23400
2350      INTEGER CHARA,ANSWR,YES,NO
2360      INTEGER BIDNUM,BIDTYP,BIDOR
2370      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
2380      INTEGER BASKET,BASK1
2390      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
2400      INTEGER CHRNAM,CHNUM,CHRBSK
2410      COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRBSK(12,300)
2420      DATA YES/'Y'/,NO/'N'/
2430      IF(NCHAR.EQ.1) GO TO 9
2440      DO 1 CHARA=1,NCHAR
2450      1  CHNUM(CHARA)=0
2460      DO 2 NBID=1,NBIDS
2470      INDEX=BIDTYP(NBID)
2480      CHNUM(INDEX)=CHNUM(INDEX)+1
2490      2  CHRBSK(INDEX,CHNUM(INDEX))=NBID
2500      DO 3 CHARA=1,NCHAR
2510      IF(CHNUM(CHARA).GE.NBASK) GO TO 3
2520      WRITE(" ",3) CHNAM(CHARA,1),CHNAM(CHARA,2),CHNUM(CHARA),NBASK
2530      3  FORMAT(' ** WARNING ** THE NUMBER OF BIDS OF THE TYPE, ',2A4,', '
2540      -,/,', TOTALS ONLY, ',I3,', WHICH IS LESS THAN'
2550      -,/,', THE NUMBER OF BASKETS, ',I3,', TO BE FORMED.')
2560      WRITE(" ",4)
2561      PRINT,""
2570      4  FORMAT(' DO YOU WISH TO CONTINUE? (YES OR NO)')
2580      5  READ(" ",6) ANSWR
2590      6  FORMAT(A1)
2600      IF(ANSWR.EQ.YES) GO TO 8
2610      IF(ANSWR.EQ.NO) STOP
2620      WRITE(" ",7)
2621      PRINT,""
2630      7  FORMAT('REENTER "YES OR NO"')
2640      GO TO 5
2650      8  CONTINUE
2660      RETURN
2670      9  CHNUM(1)=NBIDS
2680      DO 10 NBID=1,NBIDS
2690      10  CHRBSK(1,NBID)=NBID
2700      RETURN
2710      END
27200

```

```

2730C
2740      SUBROUTINE BSKORD
2750C      PUTS BIDS WITHIN EACH CHARACTER TYPE BASKET IN ORDER
2760C      FROM LARGEST TO SMALLEST.
2770C
2780      INTEGER CHARA,FIRST,RECENT,TEMP
2790      INTEGER BIDNUM,BIDTYP,BIDOR
2800      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
2810      INTEGER CHRNAM,CHNUM,CHRSK
2820      COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRESK(12,300)
2830      COMMON /PLACE1/IND1
2840      IND1=NCHAR
2850      DO 5 CHARA=1,NCHAR
2860      RECENT=1
2870      LAST=CHNUM(CHARA)
2880      1 FIRST=RECENT+1
2890      DO 2 J=FIRST, LAST
2900      JM1=J-1
2910      IF(BIDPRI(CHRSK(CHARA,JM1)).GE.BIDPRI(CHRESK(CHARA,J))) GO TO 2
2920      RECENT=JM1
2930      TEMP=CHRESK(CHARA,JM1)
2940      CHRSK(CHARA,JM1)=CHRSK(CHARA,J)
2950      CHRESK(CHARA,J)=TEMP
2960      2 CONTINUE
2970      IF(RECENT+1.EQ.FIRST) GO TO 5
2980      LAST=RECENT
2990      J=LAST
3000      3 JM1=J-1
3010      IF(BIDPRI(CHRESK(CHARA,JM1)).GE.BIDPRI(CHRESK(CHARA,J))) GO TO 4
3020      RECENT=J
3030      TEMP=CHRSK(CHARA,JM1)
3040      CHRSK(CHARA,JM1)=CHRSK(CHARA,J)
3050      CHRESK(CHARA,J)=TEMP
3060      4 J=J-1
3070      IF(J.GE.FIRST) GO TO 3
3080      IF(RECENT.LT.LAST) GO TO 1
3090      5 CONTINUE
3100      RETURN
3110      END
3120C
3130C
3140      SUBROUTINE BIDORD
3150C      DETERMINES ORDER IN WHICH BIDS WILL BE PLACED IN GROUPS OF NBASK.
3160C      CREATES - BIDOR(N) - WHICH IS THE INDEX OF THE CHARACTERISTIC
3170C      OF THE NTH BID GROUPING.
3180C      NGROUP - NUMBER OF SUCH GROUPINGS.
3190C
3200      INTEGER CHARA
3210      INTEGER BIDNUM,BIDTYP,BIDOR
3220      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
3230      INTEGER BASKET,BASK1
3240      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
3250      INTEGER CHRNAM,CHNUM,CHRSK
3260      COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRESK(12,300)
3270      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX

```



```

3200      NCHAR1=NCHAR
3200      NGROUP=0
3300      NUMBID=0
3310      IF(NCHAR.EQ.1) GO TO 4
3320      DO 1 CHARA=1,NCHAR
3330      1      NCHR(CHARA)=0
3340      DO 3 N=1,400
3350      IF(NUMBID.GE.NBIDS) RETURN
3360      CALL MAXAVG
3370      NGROUP=NGROUP+1
3380      BIDOR(NGROUP)=MAX
3390      3      CONTINUE
3400      RETURN
3410      4      NGROUP=NBIDS/NBASK
3420      IF(NBASK*NGROUP.NE.NBIDS) NGROUP=NGROUP+1
3430      DO 5 NGROUP=1,NGROUP
3440      5      BIDOR(NGROUP)=1
3450      NCHR(1)=NBIDS
3460      RETURN
3470      END
3480C
3490C
3500      SUBROUTINE MAXAVG
3510C      CALCULATES AVERAGE OF THE NEXT NBASK BIDS OF EACH CHARACTERISTIC
3520C      TYPE, AND RETURNS THE INDEX OF THE CHARACTERISTIC TYPE HAVING THE
3530C      LARGEST AVERAGE.
3540C      NCHR(N) - INDEX OF POSITION TO BEGIN AVERAGE OF BIDS FOR
3550C      CHARACTERISTIC TYPE N.
3560C
3570      INTEGER START,END
3580      INTEGER CHRNAM,CHNUM,CHRSK
3590      INTEGER BIDNUM,BIDTYP,BIDOR
3600      INTEGER BASKET,BASK1
3610      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),ETOT(10),NGROUP
3620      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
3630      COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRSK(12,300)
3640      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
3650      MAX=1
3660      DO 2 N=1,NCHAR1
3670      .VG(N)=0
3680      START=NCHR(N)+1
3690      END=NCHR(N)+NBASK
3700      IF(END.GT.CHNUM(N)) END=CHNUM(N)
3710      IF(START.GT.END) GO TO 2
3720      DO 1 J=START,END
3730      1      AVG(N)=AVG(N)+BIDPRI(CHRSK(N,J))
3740      IF(START.EQ.END) GO TO 3
3750      AVG(N)=AVG(N)/(END-START)
3760      2      IF(AVG(N).LT.AVG(MAX)) GO TO 2
3770      MAX=N
3780      MAXBIDS=END-START+1
3790      2      CONTINUE
3800      NUMBID=NUMBID+MAXBIDS
3810      NCHR(MAX)=NCHR(MAX)+MAXBIDS
3820      RETURN
3830      END
3840C

```

```

2850C
2860C
2870C      SUBROUTINE BASKT
2880C      PLACES BIDS INTO BASKETS ACCORDING TO BIDOR AND BASKET TOTALS.
2890C      BTOT(N) - TOTAL OF BIDS IN BASKET N.
2900C
2910C      REAL HIDE(10)
2920C      INTEGER START(12),UP1(10),ORJ,ORDER1(10),ORDER2(10),BIDORI
2930C      INTEGER CHRNAM,CHNRUM,CHRESK
2940C      INTEGER BIDNUM,BIDTYP,BIDOR
2950C      INTEGER BASKET,BASK1
2960C      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
2970C      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
2980C      COMMON /CHRCTR/ NCHAR,CHNRUM(12,2),CHNRUM(12),CHRESK(12,300)
2990C      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
4000C      DO 1 N=1,NBASK
4010C          HIDE(N)=0
4020C          BTOT(N)=0
4030C      1  UP1(N)=0
4040C          DO 20 N=1,NCHAR
4050C      20  START(N)=0
4060C          DO 6 I=1,NGROUP
4070C              BIDORI=BIDOR(I)
4080C              NUMLEF=NCHR(BIDORI)-START(BIDORI)
4090C              CALL ORDER(ORDER1,BTOT,NBASK)
4100C              IF(NUMLEF.LT.NBASK) GO TO 3
4110C              DO 2 J=1,NBASK
4120C                  ORJ=ORDER1(J)
4130C                  INDEX=CHRESK(BIDORI,START(BIDORI)+J)
4140C                  PRICE=BIDPRI(INDEX)
4150C                  BASKET(ORJ,I)=INDEX
4160C      2  BTOT(ORJ)=BTOT(ORJ)+PRICE
4170C              START(BIDORI)=START(BIDORI)+NBASK
4180C              GO TO 6
4190C      3  CALL UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
4200C              AVGE=0.
4210C              DO 4 J=1,NUMLEF
4220C                  ORJ=ORDER2(J)
4230C                  INDEX=CHRESK(BIDORI,START(BIDORI)+J)
4240C                  PRICE=BIDPRI(INDEX)
4250C                  BASKET(ORJ,I)=INDEX
4260C                  AVGE=AVGE+PRICE
4270C      4  BTOT(ORJ)=BTOT(ORJ)+PRICE
4280C              AVGE=AVGE/NUMLEF
4290C              NUMLEF1=NUMLEF+1
4300C              DO 5 J=NUMLEF1,NBASK
4310C                  ORJ=ORDER2(J)
4320C

```

```

4030C WHEN NUMBER OF BIDS LEFT OF ANY PARTICULAR CHARACTERISTIC TYPE
4040C IS LESS THAN NUMBER OF BASKETS, THE AVERAGE OF THESE BIDS IS
4050C CALCULATED, THE BIDS ARE PLACED INTO THE BASKETS, AND THE
4060C REMAINING UNFILLED BASKETS RECEIVE THE AVERAGE OF THE BIDS.
4070C (SINCE BASKET(N,K) CONTAINS ONLY INDICES OF PROPOSALS AND NOT
4080C ACTUAL BID PRICES, THE UNFILLED BASKETS RECEIVE THE VALUE -1
4090C INTO BASKET(N,K).)
4100C     HIDE(N) - ACCUMULATES THE PHANTOM AVERAGES IN BASKET N
4110C     FOR LATER CORRECTION OF BTOT(N).
4120C
4130C     BASKET(ORJ,I)=-1
4140C     HIDE(ORJ)=HIDE(ORJ)+AVGE
4150C 5     BTOT(ORJ)=BTOT(ORJ)+AVGE
4160C 6     CONTINUE
4170C     DO 7 J=1,NBASK
4180C 7     BTOT(J)=BTOT(J)-HIDE(J)
4190C     RETURN
4200C     END
4210C
4220C     SUBROUTINE ORDER(ORDER1,BTOT,NBASK)
4230C     ORDERS THE BASKET INDICES INTO - ORDER1 - FROM SMALLEST BASKET
4240C     TO LARGEST BASKET.
4250C
4260C     INTEGER ORDER1(10)
4270C     REAL BTOT(10)
4280C     DO 1 N=1,NBASK
4290C 1     ORDER1(N)=N
4300C     NBASK1=NBASK-1
4310C     DO 3 N=1,NBASK1
4320C     MIN=N
4330C     NBASK2=N+1
4340C     DO 2 K=NBASK2,NBASK
4350C     IF(BTOT(ORDER1(MIN)).LE.BTOT(ORDER1(K))) GO TO 2
4360C     ITEMP=ORDER1(MIN)
4370C     ORDER1(MIN)=ORDER1(K)
4380C     ORDER1(K)=ITEMP
4390C 2     CONTINUE
4400C 3     CONTINUE
4410C     RETURN
4420C     END
4430C
4440C     SUBROUTINE UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
4450C     DETERMINES WHICH BASKETS HAVE ONE MORE BID THAN OTHERS AND RETURNS
4460C     - ORDER2 - THE NEW ORDER OF BASKET INDICES REFLECTING SMALLEST TO
4470C     LARGEST AND NUMBER OF BIDS IN BASKETS.
4480C
4490C     INT. GER ORDER1(10),ORDER2(10),UP1(10),ENDJ,STARTJ
4500C     STARTJ=0
4510C     DO 1 I=1,NBASK
4520C     IF(UP1(ORDER1(I)).EQ.1) GO TO 1
4530C     STARTJ=STARTJ+1
4540C     ORDER2(STARTJ)=ORDER1(I)
4550C 1     CONTINUE

```

```

4880      IF(STARTJ.NE.0) GO TO 3
4890      DO 2 J=1,NBASK
4900      2  ORDER2(J)=ORDER1(J)
4910      GO TO 3
4920      3  IF(STARTJ.EQ.NBASK) GO TO 8
4930      ENDJ=STARTJ
4940      DO 4 I=1,NBASK
4950      IF(UP1(ORDER1(I)).EQ.0) GO TO 4
4960      ENDJ=ENDJ+1
4970      ORDER2(ENDJ)=ORDER1(I)
4980      4  CONTINUE
4990      5  IF(STARTJ.GE.NUMLEF) GO TO 10
5000      DO 6 J=1,NUMLEF
5010      IF(UP1(ORDER2(J)).EQ.0) UP1(ORDER2(J))=0
5020      6  IF(UP1(ORDER2(J)).EQ.1) UP1(ORDER2(J))=1
5030      NMLEF1=NUMLEF+1
5040      DO 7 J=NMLEF1,NBASK
5050      7  UP1(ORDER2(J))=0
5060      RETURN
5070      8  NMLEF1=NUMLEF+1
5080      DO 9 J=NMLEF1,NBASK
5090      9  UP1(ORDER2(J))=0
5100      10 DO 11 J=1,NUMLEF
5110      11 UP1(ORDER2(J))=1
5120      RETURN
5130      END
5140C
5150C
5160      SUBROUTINE SWAP
5170C      PERFORMS A BID SWAP BETWEEN BASKETS TO BALANCE BASKET TOTALS.
5180C      LGESK - INDEX OF LARGEST BASKET WITH RESPECT TO BASKET TOTALS.
5190C      SMESK - INDEX OF SMALL BASKDET.
5200C
5210      INTEGER ORDER1(10),ESK,SWTCH1,SWTCH2
5220      INTEGER BIDNUM,BIDTYP,BIDOR
5230      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
5240      INTEGER BASKET,BASK1
5250      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
5260      INTEGER CHRNAM,CHNUM,CHRESK
5270      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHNUM(12),CHRBSK(12,300)
5280      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
5290      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
5300      CALL SETUP
5310      1  CALL ORDER(ORDER1,BTOT,NBASK)
5320      DO 3 N=2,NBASK
5330      LGBSK=ORDER1(NBASK-N+2)
5340      NBASK1=NBASK-1
5350      DO 2 K=1,NBASK1
5360      SMESK=ORDER1(K)
5370      DIFF=BTOT(LGBSK)-BTOT(SMESK)
5380      IF(DIFF.EQ.0.) GO TO 3

```

```

5390      CALL DIFFNT(SWTCM2,DIFF)
5400      IF(SMIND.NE.0) GO TO 4
5410      2   CONTINUE
5420      3   CONTINUE
5430      RETURN
5440      4   DIFF=BIDPRI(BASK1(LGBSK,LGIND))-BIDPRI(BASK1(SMBSK,SMIND))
5450      BTOT(LGBSK)=BTOT(LGBSK)-DIFF
5460      BTOT(SMBSK)=BTOT(SMBSK)+DIFF
5470      CALL PLACE
5480      GO TO 1
5490      END
5500C
5510C
5520      SUBROUTINE SETUP
5530C      INITIALIZES - BASK1(N,K) - TO BE THE SAME AS - BASKET(N,K) - BUT
5540C      WITH BIDS PLACED TOGETHER WITHIN BASKETS BY CHARACTERISTIC TYPE.
5550C      NOTE: BIDS WILL ALSO BE IN ORDER FROM LARGEST TO SMALLEST
5560C      WITHIN CHARACTER GROUPINGS SINCE BASKET WAS ORDERED.
5570C
5580      INTEGER COUNT(12),POS,BSK
5590      INTEGER BIDNUM,BIDTYP,BIDR
5600      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDR(160)
5610      INTEGER BASKET,BASK1
5620      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
5630      INTEGER CHRNAM,CHNUM,CHRSK
5640      COMMON /CHRCR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRSK(12,300)
5650      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
5660      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
5670      DO 1 N=1,NCHAR
5680      1   COUNT(N)=0
5690      DO 2 N=1,NGROUP
5700      NUMB=BIDR(N)
5710      2   COUNT(NUMB)=COUNT(NUMB)+1
5720      STARTC(1)=1
5730      DO 3 N=2,NCHAR
5740      STARTC(N)=COUNT(N-1)+STARTC(N-1)
5750      3   COUNT(N-1)=0
5760      COUNT(NCHAR)=0
5770      DO 5 N=1,NGROUP
5780      NUMB=BIDR(N)
5790      POS=STARTC(NUMB)+COUNT(NUMB)

```

```

5120      DO 3 BCK=1,NBACK
5130      BASK1(LCK,PBC)=BASKET(BCK,N)
5140      COUNT(NUMF)=COUNT(NUME)+1
5150      RETURN
5160      END
5170
5180      SUBROUTINE DIFFNT(SWCH2,DIFF)
5190      DETERMINES IF/WHICH BID SWAPS FOR GIVEN BASKETS WILL PRODUCE A
5200      BETTER BALANCE.
5210      LGIND - INDEX OF BID TO BE SWAPPED OUT OF LARGE BASKET.
5220      SMIND - INDEX OF BID TO BE SWAPPED OUT OF SMALL BASKET.
5230
5240      INTEGER SWCH1,SWCH2,STRT,STP,CHARA
5250      INTEGER BIDNUM,BIDTYP,BIDOR
5260      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
5270      INTEGER BASKET,BASK1
5280      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTCT(10),NGROUP
5290      INTEGER CHRNAM,CHRNUM,CHRESK
5300      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRESK(12,300)
5310      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
5320      COMMON /SWAP1/ LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
5330      SMIND=0
5340      DIFF1=0
5350      DO 3 CHARA=1,NCHAR
5360      STRT=STARTC(CHARA)
5370      STP=NGROUP
5380      IF(CHARA.LT.NCHAR)STP=STARTC(CHARA+1)-1
5390      DO 2 SWCH1=STRT,STP
5400      IND1=BASK1(LGBSK,SWCH1)
5410      IF(IND1.EQ.-1) GO TO 2
5420      PRICE1=BIDPRI(IND1)
5430      DO 1 SWCH2=STRT,STP
5440      IND2=BASK1(SMBSK,SWCH2)
5450      IF(IND2.EQ.-1.) GO TO 1
5460      PRICE2=BIDPRI(IND2)
5470      DIFF2=PRICE1-PRICE2
5480      IF(DIFF2.LE.0.) GO TO 1
5490      IF(DIFF2.GE.DIFF) GO TO 2
5500      IF(ABS(DIFF2-DIFF/2.).GE.ABS(DIFF1-DIFF/2.)) GO TO 2
5510      DIFF1=DIFF2
5520      LGIND=SWCH1
5530      SMIND=SWCH2
5540      CHAR=CHARA
5550      IF(DIFF2.EQ.DIFF/2.) RETURN
5560      1 CONTINUE
5570      2 CONTINUE
5580      3 CONTINUE
5590      RETURN
5600      END
5610
5620
5630
5640
5650
5660
5670
5680
5690
5700
5710
5720
5730
5740
5750
5760
5770
5780
5790
5800
5810
5820
5830
5840
5850
5860
5870
5880
5890
5900
5910
5920
5930
5940
5950
5960
5970
5980
5990
6000
6010
6020
6030
6040
6050
6060
6070
6080
6090
6100
6110
6120
6130
6140
6150
6160
6170
6180
6190
6200
6210
6220
6230
6240
6250
6260
6270
6280
6290
6300
6310
6320
6330
6340
6350
6360
6370
6380
6390
6400
6410
6420
6430
6440
6450
6460
6470
6480
6490
6500
6510
6520
6530
6540
6550
6560
6570
6580
6590
6600
6610
6620
6630
6640
6650
6660
6670
6680
6690
6700
6710
6720
6730
6740
6750
6760
6770
6780
6790
6800
6810
6820
6830
6840
6850
6860
6870
6880
6890
6900
6910
6920
6930
6940
6950
6960
6970
6980
6990
7000

```

```

04100
04200
04300 SUBROUTINE PLACE
04400 SWAPS BIDS.
04500
04600 INTEGER LGBSK, LGIND, SMBSK, SMIND, STARTC, CHAR
04700 COMMON /SWAP1/ LGPSK, LGIND, SMBSK, SMIND, STARTC(12), CHAR
04800 INTEGER BIDNUM, BIDTYP, BIDOR
04900 COMMON /BID/ NBIDS, BIDNUM(300), BIDPRI(300), BIDTYP(300), BIDOR(160)
05000 INTEGER BASKET, BASK1
05100 COMMON /BASK/ NBASK, BASKET(10, 300), BASK1(10, 160), ETOT(10), NGRUP
05200 TEMP=BASK1(LGBSK, LGIND)
05300 BASK1(LGBSK, LGIND)=BASK1(SMBSK, SMIND)
05400 BASK1(SMBSK, SMIND)=TEMP
05500 CALL PUT(LGBSK, LGIND, CHAR)
05600 CALL PUT(SMBSK, SMIND, CHAR)
05700 RETURN
05800 END
05900
06000
06100 SUBROUTINE PUT(BSK, IND, CHAR1)
06200 MOVES 'NEW BID' INTO PROPER ORDERED POSITION WITHIN CHARACTER TYPE
06300 GROUPING WITHIN BASKET.
06400
06500 INTEGER BIDNUM, BIDTYP, BIDOR
06600 COMMON /BID/ NBIDS, BIDNUM(300), BIDPRI(300), BIDTYP(300), BIDOR(160)
06700 INTEGER BASKET, BASK1
06800 COMMON /BASK/ NBASK, BASKET(10, 300), BASK1(10, 160), ETOT(10), NGRUP
06900 INTEGER CHRNAM, CHNUM, CHRESK
07000 COMMON /CHRCTR/ NCHAR, CHRNAM(12, 2), CHNUM(12), CHRESK(12, 300)
07100 INTEGER LGBSK, LGIND, SMBSK, SMIND, STARTC, CHAR
07200 COMMON /SWAP1/ LGPSK, LGIND, SMBSK, SMIND, STARTC(12), CHAR
07300 INTEGER BSK, CHAR1, STRT, STP
07400 IF(IND.EQ.STARTC(CHAR1)) GO TO 2
07500 IF(BIDPRI(BASK1(BSK, IND)).LE.BIDPRI(BASK1(BSK, IND-1))) GO TO 2
07600 STRT=STARTC(CHAR1)
07700 STP=IND-1
07800 DO 1 K=STRT, STP
07900 KK=STP+STRT-K+1
08000 KK1=KK-1
08100 IF(BIDPRI(BASK1(BSK, KK)).LE.BIDPRI(BASK1(BSK, KK1))) RETURN
08200 TEMP=BASK1(BSK, KK)
08300 BASK1(BSK, KK)=BASK1(BSK, KK1)
08400 BASK1(BSK, KK1)=TEMP
08500 1 CONTINUE
08600 RETURN
08700
08800 2 STRT=IND
08900 IF(STRT.EQ.NGRUP) RETURN
09000 STP=NGRUP-1
09100 IF(CHAR1.LT.NCHAR) STP=STARTC(CHAR1+1)-2
09200 IF(STRT.GT.STP) RETURN
09300 DO 3 K=STRT, STP
09400 IF(BASK1(BSK, K+1).EQ.-1) RETURN
09500 IF(BIDPRI(BASK1(BSK, K)).GE.BIDPRI(BASK1(BSK, K+1))) RETURN
09600 TEMP=BASK1(BSK, K)
09700 BASK1(BSK, K)=BASK1(BSK, K+1)
09800 BASK1(BSK, K+1)=TEMP
09900 3 CONTINUE
10000 RETURN
10100 END

```

```

0010      SUBROUTINE STATS
0020      CALL POTERE
0030      CALL MOMENTS
0040      RETURN
0050      END

0060      SUBROUTINE POTERE
0070      CALCULATES PER CENT ERROR BOUNDS.

0080      INTEGER BSK,ORDER1(10)
0090      REAL LOWEND,MAX,MIN,MAXDIF
0100      COMMON /ERRBND/ MAXDIF,LOWEND,UPREND
0110      INTEGER BASKET,BASK1
0120      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,100),BTOT(10),NGROUP
0130      CALL ORDER(ORDER1,BTOT,NBASK)
0140      MAX=BTOT(ORDER1(NBASK))
0150      MIN=BTOT(ORDER1(1))
0160      MAXDIF=MAX-MIN
0170      TOTAL=0.
0180      DO 1 BSK=1,NBASK
0190      1 TOTAL=TOTAL+BTOT(BSK)
0200      LOWEND=100.*(TOTAL/NBASK/MAX-1.)
0210      UPREND=100.*(TOTAL/NBASK/MIN-1.)
0220      RETURN
0230      END

0240      SUBROUTINE MOMENTS
0250      CALCULATES MEAN, STD DEV FOR EACH BASKET AND POPULATION.

0260      INTEGER BSK
0270      COMMON /MOM/ BSKMN(10),BSKVAR(10),POPMN,POPVAR,NBASK1(10),POPTOT
0280      INTEGER BASKET,BASK1
0290      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,100),BTOT(10),NGROUP
0300      INTEGER BIDNUM,BIDTYP,BIDOR
0310      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOP(100)
0320      DO 1 BSK=1,NBASK
0330      1 BSKVAR(BSK)=0.
0340      POPMN=0.
0350      POPVAR=0.
0360      DO 3 BSK=1,NBASK
0370      NEID=0
0380      DO 2 N=1,NGROUP
0390      INDEX=BASK1(BSK,N)
0400      IF(INDEX.EQ.-1) GO TO 2
0410      NEID=NEID+1
0420      BSKVAR(BSK)=BSKVAR(BSK)+BIDPRI(INDEX)**2
0430      2 CONTINUE
0440      POPMN=POPMN+BTOT(BSK)
0450      POPVAR=POPVAR+BSKVAR(BSK)
0460      BSKMN(BSK)=BTOT(BSK)/NEID
0470      BSKVAR(BSK)=SQRT((BSKVAR(BSK)-BTOT(BSK)**2/NEID)/(NEID-1))
0480      NBASK1(BSK)=NEID
0490      3 CONTINUE
0500      POPVAR=SQRT((POPVAR-POPMN**2/NBIDS)/(NBIDS-1))
0510      POPTOT=POPMN
0520      POPMN=POPMN/NBIDS
0530      RETURN
0540      END

```



```

7510C
7540C      SUBROUTINE OUTPUT
7550C      CALL RNSTAT
7560C      CALL LSTESK
7570C      CALL BSTATE
7580C      RETURN
7590C      END
7600C
7610C
7620C      SUBROUTINE RNSTAT
7630C      PRINTS RUN STATISTICS.
7640C
7650C      INTEGER BIDNUM,BIDTYP,BIDOR
7660C      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
7670C      INTEGER BASKET,BASK1
7680C      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
7690C      INTEGER CHRNAM,CHNUM,CHRESK
7700C      COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRESK(12,300)
7710C      WRITE(" ",1)
7720C      1  FORMAT('1',/,5X,'RUN STATISTICS')
7730C      WRITE(" ",2) NBASK
7740C      2  FORMAT(' ',/,10X,'NUMBER OF BASKETS =',5X,I3)
7750C      WRITE(" ",4) NCHAR
7760C      4  FORMAT(10X,'NUMBER OF CHARACTER TYPES =',I3)
7770C      WRITE(" ",3) NBIDS
7780C      3  FORMAT(10X,'NUMBER OF BIDS =',11X,I3)
7790C      IF(NCHAR.EQ.1) GO TO 9
7800C      WRITE(" ",5)
7810C      5  FORMAT(10X,'NUMBER OF BIDS PER CHARACTER TYPE:',/)
7820C      DO 6 N=1,NCHAR
7830C      6  WRITE(" ",7) N,CHNAM(N,1),CHNAM(N,2),CHNUM(N)
7840C      7  FORMAT(20X,I2,',',2X,2A4,2X,'=',2X,I3)
7850C      9  WRITE(" ",3)
7860C      8  FORMAT(' ',5X,40(' '))
7870C      RETURN
7880C      END
7890C
7900C      SUBROUTINE LSTESK
7910C      LISTS BASKETS.
7920C
7930C      INTEGER BSK
7940C      INTEGER BIDNUM,BIDTYP,BIDOR
7950C      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
7960C      INTEGER BASKET,BASK1
7970C      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
7980C      INTEGER CHRNAM,CHNUM,CHRESK
7990C      COMMON /CHRCTR/ NCHAR,CHNAM(12,2),CHNUM(12),CHRESK(12,300)
8000C      DO 10 BSK=1,NBASK
8010C      10  WRITE(" ",1) BSK
8020C      1  FORMAT(' ',/,5X,'BASKET NO.',2X,I2)

```

```

1040 IF(NCHAR.EQ.1) GO TO 3
1050 WRITE(" ",2)
1060 2 FORMAT(' ',/,'1X','PROPOSAL NO.','1X','BID PRICE')
1070 GO TO 5
1080 1 WRITE(" ",4)
1090 4 FORMAT(' ',12X,'PROPOSAL NO.','10X','BID PRICE',10X,'CHARACTERIST',
1100 'IC',/)
1110 MINUS=0
1120 5 DO 6 N=1,NGROUP
1130 NN=N-MINUS
1140 INDEX=BASK1(BSK,N)
1150 IF(INDEX.EQ.-1) GO TO 3
1160 IF(NCHAR.EQ.1) GO TO 7
1170 WRITE(" ",6)NN,BIDNUM(INDEX),BIDPRI(INDEX),CHRNAM(BIDTYP(INDEX),1),
1180 ACHRNAM(BIDTYP(INDEX),2)
1190 6 FORMAT(T10,I3,'.',T17,I4,T25,F10.3,T28,2A4)
1200 GO TO 3
1210 9 MINUS=MINUS+1
1220 GO TO 8
1230 7 WRITE(" ",6) N,BIDNUM(INDEX),BIDPRI(INDEX)
1240 8 CONTINUE
1250 10 CONTINUE
1260 WRITE(" ",11)
1270 11 FORMAT(' ',5X,'40('-''))
1280 RETURN
1290 END

```

# SUBROUTINE BSTATS PRINTS BASKET STATISTICS.

```

1300 INTEGER BIDNUM,BIDTYP,BIDOR
1310 COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
1320 INTEGER BASKET,BASK1
1330 COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
1340 INTEGER CHRNAM,CHRNUM,CHRESK
1350 COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRESK(12,300)
1360 REAL LOWEND,MAX,MIN,MAXDIF
1370 COMMON /ERREND/ MAXDIF,LOWEND,UPREND
1380 COMMON /MCM/ BSKNM(10),BSKVAR(10),POPVM,POPVAR,NBASK1(10),POPTOT
1390 WRITE(" ",1)
1400 1 FORMAT(' ',/,'5X','BASKET STATISTICS',/)
1410 WRITE(" ",2) MAXDIF
1420 2 FORMAT(T11,'DIFFERENCE BETWEEN MAX AND MIN BASKETS =',2X,F10.3)
1430 WRITE(" ",3) LOWEND,UPREND
1440 3 FORMAT(T11,'MEAN ERROR BOUNDED BY',F5.2,' AND ',F5.2,'.')
1450 WRITE(" ",4)
1460 4 FORMAT(T11,'BID PRICE MOMENTS:')
1470 WRITE(" ",5)
1480 5 FORMAT(' ',/,'T13','BASKET',T23,'NO. PROPOSALS',T40,'TOTAL OF BIDS',
1490 'T60','MEAN',T70,'STD DEV')
1500 DO 6 N=1,NBASK
1510 6 WRITE(" ",7) N,NBASK1(N),BTOT(N),BSKNM(N),BSKVAR(N)
1520 7 FORMAT(T14,I2,T29,I3,T42,F10.3,T55,F10.3,T66,F10.3)
1530 WRITE(" ",8) NBIDS,POPTOT,POPVM,POPVAR
1540 8 FORMAT(' ',T13,'POPUL',T20,I3,T42,F10.3,T55,F10.3,T66,F10.3)
1550 RETURN
1560 END

```

```

0610
0620 SUBROUTINE CONT(*,*)
0630 INTEGER ANSWR,YES,NO
0640 DATA YES/'Y'/,NO/'N'/
0650 WRITE(" ",1)
0660 PRINT,""
0670 1 FORMAT( /,' DO YOU WISH TO SWAP BIDS BETWEEN! SELECTED BASKETS?',
0680 2 '(YES OR NO)')
0690 2 READ(" ",7) ANSWR
0700 IF(ANSWR.NE.YES) GO TO 3
0710 CALL SWAP2
0720 RETURN2
0730 3 IF(ANSWR.EQ.NO) GO TO 4
0740 WRITE(" ",9)
0750 PRINT,""
0760 GO TO 2
0770 4 WRITE(" ",5)
0780 PRINT,""
0790 5 FORMAT(' DO YOU WISH TO CHANGE THE NUMBER OF BASKETS? (YES OR NO)')
0800 6 READ(" ",7) ANSWR
0810 7 FORMAT(A1)
0820 IF(ANSWR.NE.YES) GO TO 3
0830 CALL NUMESK
0840 RETURN1
0850 8 IF(ANSWR.EQ.NO) STOP
0860 WRITE(" ",9)
0870 PRINT,""
0880 9 FORMAT(' REENTER YES OR NO.')
0890 GO TO 6
0900 END
0900C
0910C
0920 SUBROUTINE SWAP2
0930C PERFORMS MANUAL SWAPS.
0940C
0950 INTEGER BIDNUM,BIDTYP,BIDOR
0960 COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
0970 INTEGER BASKET,BASK1
0980 COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
0990 INTEGER ANSWR,YES,NO,NESK(2),NBID(2),IND(2),N1(2)
1000 DATA YES/'Y'/,NO/'N'/
1010 1 WRITE(" ",2)
1020 PRINT,""
1030 2 FORMAT(' INPUT: BASKET NUMBER, PROPOSAL NUMBER, BASKET NUMBER, ',
1040 3 'PROPOSAL NUMBER.')
1050 READ(" ",*) NBESK(1),NBID(1),NESK(2),NBID(2)
1060C
1070C CHECK IF BASKET NUMBERS AND PROPOSAL NUMBERS ARE VALID.
1080C

```

```

0100      DO 3 I=1,2
0110      IF(1.LE.NBSK(I).AND.NBSK(I).LE.NBACK) GO TO 4
0120      WRITE("      ",3) NBSK(I)
0130      3 FORMAT(' *** ERROR ***',1X,I4,' IS NOT A VALID BASKET NUMBER.')
0140      GO TO 13
0150      4 IF(1.LE.NBID(I).AND.NBID(I).LE.NBIDS) GO TO 5
0160      WRITE("      ",5) NBID(I)
0170      5 FORMAT(' *** ERROR *** ',I5,' IS NOT A VALID PROPOSAL NUMBER.')
0180      GO TO 13
0190      6 CONTINUE
0200
0210      SEARCH BIDNUM(N) TO FIND APPROPRIATE INDICES.
0220
0230      DO 8 I=1,2
0240      DO 7 INDEX=1,NBIDS
0250      IF(BIDNUM(INDEX).EQ.NBID(I)) GO TO 8
0260      7 CONTINUE
0270      8 IND(I)=INDEX
0280      DO 11 I=1,2
0290      DO 9 N=1,NGROUP
0300      IF(BASK1(NBSK(I),N).EQ.IND(I)) GO TO 11
0310      9 CONTINUE
0320      WRITE("      ",10) BIDNUM(IND(I)),NBSK(I)
0330      10 FORMAT(' *** ERROR *** PROPOSAL ',I4,' IS NOT IN BASKET ',I3,'.')
0340      GO TO 13
0350      11 N1(I)=N
0360
0370      SWAP BIDS AND ADJUST BASKET TOTALS.
0380
0390      DO 12 I=1,2
0400      BASK1(NBSK(I),N1(I))=IND(3-I)
0410      12 BTOT(NBSK(I))=BTOT(NBSK(I))-BIDPRI(IND(I))+BIDPRI(IND(3-I))
0420      13 WRITE("      ",14)
0430      PRINT,""
0440      14 FORMAT(' DO YOU WISH TO REENTER SWAP OR ENTER MORE SWAPS? ',
0450      &1('YES OR NO'))
0460      15 READ("      ",16) ANSWR
0470      16 FORMAT(A1)
0480      IF(ANSWR.EQ.YES) GO TO 1
0490      IF(ANSWR.EQ.NO) RETURN
0500      WRITE("      ",17)
0510      PRINT,""
0520      17 FORMAT(' REENTER YES OR NO.')
0530      GO TO 15
0540      END

```

\*\*\*\*\* 1: INPUT 01/11/71

```
101      REEL BID(100),NEGOTD(150)
110      INTEGER PROPSN(150),RETURN
120      LOGICAL QUIT
130      FILENAME BIDFIL,NEGFIL
140      PAINT 10
151      10 FORMAT("-",10("X")/" ",70(" "),10("X")/" ",20X,"PRICE COMPUTATIONS",
160      20X,10("X")/" ",26X,"BASED ON NEGOTIATED SAMPLE",20X,10("X")/" ",70X,
170      10("X")/" ",50("X")/" ")
180      RETURN=2
191      20 PRINT 30
200      30 FORMAT("CENTER NAME OF FILE CONTAINING PROPOSAL NUMBERS ",
210      10X,"AND NEGOTIATED PRICES ")
220      READ #0,NEGFIL
230      OPEN(NEGFIL,ERR=110,STATUS="OLD")
240      READ (NEGFIL,#) BIDFIL
250      40 FORMAT(10X)
260      RETURN=1
270      OPEN(BIDFIL,ERR=110,STATUS="OLD")
280      DO 50 I=1,300
290      50 READ (BIDFIL,#,END=70,ERR=110) BID(I)
300      I=301
310      70 NEID=I-1
320      CLOSE(BIDFIL)
330      RETURN=2
340      DO 100 I=1,150
350      100 READ (NEGFIL,#,END=150,ERR=110) PROPSN(I),NEGOTD(I)
360      I=151
370      GO TO 150
380      110 I=IERROR()
390      GO TO (120,140),RETURN
400      120 PRINT 130,BIDFIL,I
410      130 FORMAT("ERROR ENCOUNTERED ON ",A9," FILE. FAILURE CODE IS",I9,".")
420      GO TO 20
430      140 PRINT 130,NEGFIL,I
440      GO TO 20
450      150 NNEG=I-1
460      CLOSE NEGFIL
470      IORDER=NNEG
480      DO 170 K=2,NNEG
490      170 QUIT=.TRUE.
500      DO 160 I=2,IORDER
510      160 IM1=I-1
520      IF (PROPSN(IM1).LE.PROPSN(I)) GO TO 160
530      QUIT=.FALSE.
540
```

```

      HOLD=PROPSN(I)
      PROPSN(I)=PROPSN(IM1)
      PROPSN(IM1)=HOLD
      HOLD=NEGOTD(I)
      NEGOTD(I)=NEGOTD(IM1)
      NEGOTD(IM1)=HOLD
      CONTINUE
      IF (NBIT) GO TO 100
      170 ORDER=ORDER-1
      180 TNEG=0.
      190 TTRID=0.
      DO 100 I=1,NNEG
      200 TNEG=TNEG+NEGOTS(I)
      210 TTRID=TTRID+BID(PROPSN(I))
      220 RATIO=TNEG/TTRID
      PRINT 200
      230 FORMAT("PROPOSAL",12X,"BID",6X,"ESTIMATED",5X,"NEGOTIATED"/
      240 6X,"NUMBER",3(10X,"PRICE"))/" ")
      TOTRID=0.
      TOTEST=0.
      J=1
      DO 250 I=1,NBID
      260 EST=RATIO*BID(I)
      IF (PROPSN(J).EQ.I.AND.J.LE.NNEG) GO TO 220
      PRINT 210,I,BID(I),EST
      270 210 FORMAT(15,F15.3,F15.3)
      TOTEST=TOTEST+EST
      J TO 240
      280 PRINT 210,PROPSN(J),BID(I),NEGOTD(J)
      290 FORMAT(15,F15.3,15X,F15.3)
      J=J+1
      300 TOTRID=TOTRID+BID(I)
      310 CONTINUE
      PRINT 260,TOTRID,TOTEST,TNEG
      320 260 FORMAT(6X,3(6X,6(" ")))/" TOTALS",3F15.3/" ")
      330 STOP
      340 END

```

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER N134	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Basket Method for Selecting Balanced Samples - Part III: Computer Source Programs		5. TYPE OF REPORT & PERIOD COVERED Technical
		6. PERFORMING ORG. REPORT NUMBER T.R. #374
7. AUTHOR(s) K.T. Wallenius Stephen L. Benz		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-0451
9. PERFORMING ORGANIZATION NAME AND ADDRESS Clemson University Dept. of Mathematical Sciences Clemson, South Carolina 29631		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR 365-049
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Code 434 Arlington, Va. 22217		12. REPORT DATE 30 Nov. 1981
		13. NUMBER OF PAGES 58
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)  Balanced sampling.		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)  In this third of a series of three documents describing the basket method of sampling, source computer codes are provided for the convenience of users who may want to modify the programs to meet special local needs.		

DD FORM 1473

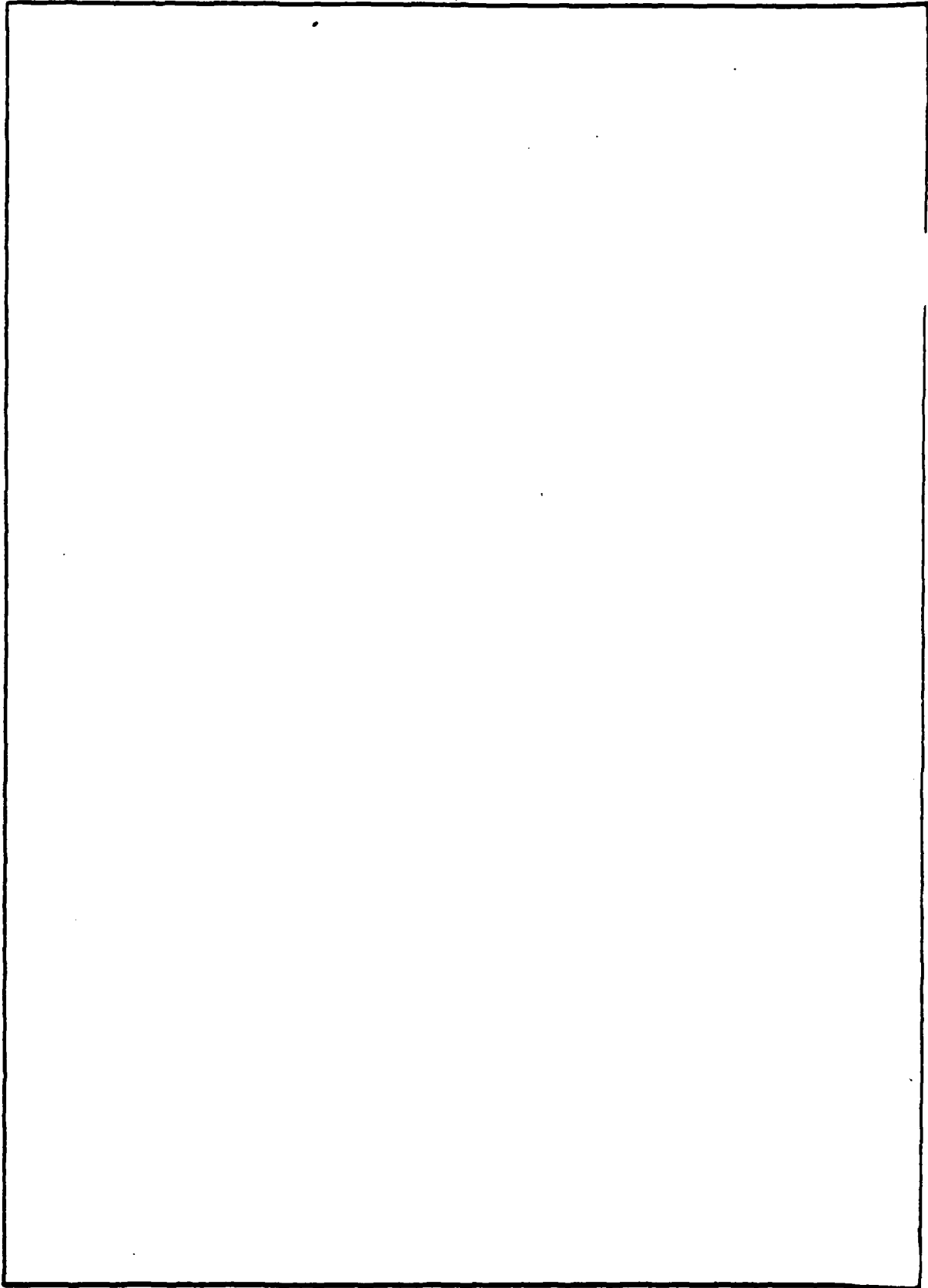
JAN 73

EDITION OF 1 NOV 65 IS OBSOLETE  
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)



SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)



